THE PENTEST EXPERTS.

WHITE PAPER

PERFORMANCE AND POSSIBLE ARRANGEMENTS OF PENETRATION TESTS

Latest

- Red teaming assessments
- Agile pentesting
- Cloud
- Embedded security
- Operational technology

Schaffhausenstraße 77, 72072 Tübingen, Germany
+49 (0)7071 - 40 78 56-0
info@syss.de
www.syss.de

Sebastian Schreiber

About the Managing Director

1993–1999	Degree in Computer Science, Physics and Mathematics and Business Administration at Eberhard Karls University in Tübingen
1996–1998	Employee at Hewlett Packard
1996	MicroGold (USA)
1998–present	Managing Director of SySS GmbH

Numerous publications, talks at home and abroad.

As of: January 2025

Authors
Sebastian Schreiber
Moritz Abrell
Fidelis Abt
Micha Borrmann
Philipp Buchegger
Matthias Deeg
Thomas Heumann
Franz Jahn
Johannes Lauinger
Gerhard Klostermeier
Torsten Lutz
Daniel Reutter
Steffen Tacke

Quality Assurance
Marcus Bauer
Stefanie Hütter
Dr. Julia Kerscher
Regina Nazarenko
Jonathan Schneider

Contents

1 Penetration Tests

SySS has been carrying out security tests since 1998. The findings from these tests form the basis of this white paper. SySS has gained its experience both with large multinational companies and traditional medium-sized enterprises. The recommendations in this white paper are based on the practical experience of our IT security consultants and intensive communication with our customers.

SySS regards security tests or penetration tests as an active quality control measure in IT security. This white paper will help you select the correct test objects and the suitable test methods for these objects from the range offered by SySS. It will also explain what preconditions are required to enable a test to be carried out efficiently and successfully. Special emphasis is placed on the decisions and measures which are required to ensure that the test is also perceived inside your company as a positive and useful service for all participants.

1.1 Organizational possibilities

Due to the differences in various test objects, security tests cannot be performed according to a fixed, standardized method, but have to be organized in a flexible manner. These arrangements depend on several factors:

– Systems, applications or other IT components that have to be tested (see Section 1.1.1)
– Selected test modules (see Section 1.3.2)
– Possible focal points (see Section 1.1.2)
– Regularity of the tests (see Section 1.1.2)
– Perspective from which the test is performed (see Section 1.1.4)
– Internal coordination of the test procedure (see Section 1.1.5)
– Special aspects in the test procedure (see Section 1.1.5)
– Handling of potential denial of service (DoS) (see Section 1.2.3)
– Available budget

1.1.1 Test object and test coverage

Both when performing external and internal tests, the test objects include, for example, systems which are identified by their IP addresses. The customer selects either a representative sample from all the IP addresses or every IP address is tested. During testing of web applications or web services, the test object is the particular service itself or its provided functional scope – for example a web-based application or an XML-based interface.

When testing Wi-Fi networks (wireless LANs), the test object is again the Wi-Fi infrastructure at a customer's location or selected wireless networks. The test coverage here describes, for example, the size of the campus to be examined or the number of buildings to be tested.

The test object and the test depth are taken into account during preparation of the offer. The time required here is also taken into account. Due to the variety of systems and applications which may be used for all test objects, it is difficult to make generalized statements. We therefore recommend that you directly discuss the test object with us beforehand.

In general and especially with large companies, a test does not cover the entire internal network or the complete external attack surface. Instead, a practical selection is made. As a special form, SySS may independently select samples from one or more networks.

If either the customer or SySS discovers during a test that changes could be practical, adjustments can be made in an unbureaucratic way. These changes are made in direct agreement between the consultant in charge and the customer's contact person.

1.1.2 Test depth and test frequency

The test depth is automatically calculated from the selected test object and the available time. If one of the objectives of the test is, for example, to gain an overview of a large number of systems in a comparatively short period, the test depth of an individual system is low and the search for very high risk potential takes priority. If, however, a long period of time is available for a few systems, even misconfigurations can be recorded, for example. Although these misconfigurations do not represent a direct security risk, they do not ideally exploit functional efficiency.

The focal point when studying the test object shown in the offer is defined in more detail during a discussion, which is normally held over the phone (KICKOFF, see Subsection 1.3.1 on Page 17). The general objective is to determine, within the test time window, the current security level of the test object and what security gaps represent the largest risk. The consultant in charge will normally invest more time and effort in detecting security gaps – which facilitate intrusion by third parties – rather than conducting a detailed study of errors which only represent a minimum risk.

The ultimate objectives are to acquire the most extensive overall picture of the security level of the test object, clearly stipulate risks and submit proposals for eliminating these risks. All this is recorded in an extensive final report (REPORT, see Subsection 1.3.3 on Page 19).

If the focal point of the test or the test depth has to be changed during the test, this can also be directly discussed between the consultant and the contact person. Since security tests do not follow a linear pattern, SySS offers the necessary flexibility in this respect.

Security tests cannot achieve their maximum effect on the security process if they are only performed once. This is because the measures used to eliminate detected security gaps after a test should become part of the work routine of the employees or service providers. In addition, software updates, or adding or removing modules or similar may lead to new security gaps. New attack techniques are also being developed constantly and vulnerabilities are being published time and again. These attack techniques and vulnerabilities are also relevant to an already tested test object and can only be recorded by regular tests.

In order to achieve lasting results, the security test should be fully integrated in the security process and carried out at regular intervals. Companies which attach great importance to IT security draw up test plans which extend two to three years into the future, for example:

	Q1 2025	Q2 2025	Q3 2025	Q4 2025	Q1 2026	Q2 2026	Q3 2026
Security test of systems on the Internet (IP-RANGE module)							
Testing web applications (WEBAPP module)							
Internal penetration test (LAN module)							
Wi-Fi test (WLAN module)							

The permanent change in IT networks and applications must be taken into account in this respect. The plan should be reconsidered and updated approximately every six months. The test object should be selected so that the benefits are maximized and no routine is created with a sense of indifference towards the test results. The test plan should have a long-term focus as this is the only way to identify weaknesses that occur and demonstrate professional quality management.

1.1.3 Test models (black, white and grey box)

A specific perspective is adopted (see Subsection 1.1.4 on the following page) and a certain level of knowledge of the potential attacker is assumed during a security test. If required, SySS uses the black box, white box and grey box model.

Black box model

In this model, the customer only provides a minimum amount of information about the test object. However, a black box test must never be misunderstood as a test where no information is transmitted between the tester and the customer, and where goals are selected and tested totally independently. Legal conditions do not allow external systems to be tested without the express permission of the actual operator.

Before a test is carried out, it must always be verified whether it is practical to test the system or the network from organizational and technical aspects. If selection becomes complex, perimeter detection (see Subsection 2.14.1 on Page 77) can be carried out beforehand. In this case, SySS mainly identifies test objectives independently. The results and the sample selection are discussed with the customer who approves the test and obtains the necessary licenses.

White box model

During tests in accordance with the white box model, extensive information about the test object is transmitted. It may be advantageous, for example, to simultaneously inspect the source code of security-related functions of a web application when analyzing the application.

Grey box model

SySS security tests generally follow this model. The customer provides precisely the information which is required to perform a security test efficiently. If more information is required, inquiries are sent to the customer's contact person. The information required for a test module is described in this white paper under "Cooperation by the customer" within the corresponding module section. Due to its many years of experience, SySS regards this model as the most effective approach for the most frequent test objectives.

1.1.4 Test perspectives

The different positions which a potential attacker can assume are covered by different test procedures. It can first be tested, for example, what infrastructure of the customer is actually accessible from the internet. A realistic picture of the customer's external attack surface is gained in this way (see also Subsection 2.14.1 on Page 77). This may also be regarded as an inventory measure. However, a test can also be carried out to determine the risk specifically emanating from systems in the internet that can be accessed by unauthorized users. For this purpose, these systems are subjected to an external security test (see Section 2.1 on Page 23).

A distinction must be made as to whether web applications, web services or mobile apps are being tested (see Sections 2.2, 2.3 and 2.9). The test is primarily carried out from the perspective of regular registered users of an application. The tester may also adopt the perspective of a non-registered user in order to also thoroughly test, for example, the authentication process for vulnerabilities.

The internal security test checks the company network from the perspective of the internal perpetrator (see Section 2.4 on Page 31). An internal perpetrator may also be an attacker who has managed to gain control of an individual system in the company network. During an internal test, it is normally possible to access a very large number of systems which also provide numerous services. The test procedure here must therefore also be changed frequently. For example, we concentrate on detecting serious security gaps which can easily be exploited ("low-hanging fruit"). Special subcomponents of an internal IT infrastructure can also be tested. Examples include network components, the VoIP and Unified Communication infrastructure or special application environments (Active Directory, SAP, etc.).

During Wi-Fi security tests, the perspective of an attacker is initially adopted within the range of a Wi-Fi access point. In this case, it is tested whether, for example, unauthorized use of a network is possible or whether existing connections of users may be compromised.

Since more and more products are now internet-enabled (Internet of Things), it is becoming increasingly more important to analyze hardware components and their interfaces to the internet. The objective here is to prevent the occurrence of security gaps which attackers can easily exploit (see Section 2.11 on Page 63).

SySS also offers red teaming assessments (see Chapter 3 on Page 89) where simulations are carried out very specifically in the role of attackers who not only attempt to obtain sensitive information and data using technical methods, but are also not afraid to use social engineering or personally infiltrate buildings.

1.1.5 Announced or unannounced tests – covert or transparent tests

The objective of penetration tests is to detect technical defects rather than human deficits. However, unannounced tests may often be regarded by the affected persons as human deficits. In this case, the major disadvantage for the customer is that doubts may be cast on the test results and employees are much less motivated to implement urgent security measures. The objective of a security test is therefore not normally attained, but is simply missed if it is performed unannounced.

The continuous success of SySS is due to the fact that both the customers' contact persons and their systems managers and administrators have trust rather than mistrust in a security test as a service. One key factor here is the offer that all participants can personally attend the test.

Tip by Sebastian Schreiber

Talk to all participants about planned tests. You will therefore make sure that the security test is regarded as a useful service and the results are processed efficiently.

The second key factor is a positive error culture. The weaknesses found in a penetration test should be understood as learning opportunities and chances for new approaches and further development.

Tip by Sebastian Schreiber

Do not take discovered mistakes as an opportunity to look for the "culprit". This may lead to cover-up and loss of confidence. A penetration test only unfolds its full effect against the background of a positive error culture.

In addition to all technical possibilities, the unannounced and covert method called "social engineering" is one of the most effective ways in which to acquire sensitive data. Social engineering in the context of IT security means acquiring sensitive data by deceiving a person. Fraudsters carrying out social engineering pretend, for example, to be authorized technicians or external service providers and succeed in retrieving requested information by means of their changed image. They also normally have sufficient knowledge of corporate culture to exploit hectic situations (e.g. large-scale IT changes, moves, all kinds of business and company-related situations, etc.).

Social engineering harbors an enormous risk for companies. The discussion concerning measures for limiting this potential risk is entirely justified. However, there is one major difference compared with all other test possibilities: The test object in this case is a person, not a technical component. The tests are not deterministic and are normally used to test existing processes in a company and make employees aware.

Testers assume a false identity in these tests in order to induce company employees to hand over sensitive data by means of e-mails, phone calls or other direct interaction. Outside tightly controlled conditions, this is legally critical and delicate from organizational aspects. These tests must therefore always be announced within the framework of awareness measures.

As already mentioned, security tests – except red teaming tests – are not carried out covertly. The consultant in charge does not take any special measures whatsoever to hide the test activities. Past experience shows that measures which are suitable, for example, for hiding the test from automated detection or defense systems increase the duration of the test enormously – often more than is conducive to a normal project schedule. Based on the report prepared during the documentation phase, it can also be clarified whether, for example, automatic systems recognized the tests which substantiated the security gaps with high risk potential.

1.1.6 Special procedure

Specialization

SySS specializes in carrying out security tests. This extremely high level of specialization ensures that every consultant has a wealth of experience that is continuously extended when performing the regular tests.

SySS is therefore very well aware of the needs of its customers and can provide precise test results. The company provides a critical yet extensive external perspective of the security level. Consultancy services for rectifying detected security gaps are therefore only required to a minimal extent, since our customers are very successful in handling the implementation of the necessary measures themselves.

Transparency

SySS attaches great importance to ensuring that its customers can readily understand how security gaps were recognized and exploited during the test. It is counterproductive to regard hacking as a secret and alien process if IT security has to be improved inside a company.

The necessary transparency is achieved by observing the following aspects:

– High-quality documentation is prepared with the declared objective of making security problems transparent.
– It is helpful if the customer invites his employees and service providers affected by tests to attend all or part of the security test. SySS is therefore always willing to help the customer with its knowledge.
– On request, the results will be presented by the consultant who conducted the test. If the opportunity arises, individual attacks can also be demonstrated once again within this context.

This transparency is the only way to ensure that your employees have a positive perception of security tests.

Flexibility

When performing tests, SySS works in a flexible manner rather than according to a fixed pattern. A fixed pattern would misjudge the nature of an attack since every network is different and every step during the test depends on the previous step. Practical changes in the focal points during the test can also be made through a discussion between the contact person and the consultant in charge.

Reaction speed

With the agile team, SySS offers ad hoc penetration tests. At the customer's request, the project can start on the same day the order is placed (kick-off meeting), and the project can then be conducted a few days later. Team agile can be used for all test modules that do not require any personal accounts or a fixed period of time planned in advance and which, in the case of tests on internal networks, can be carried out with the pentest box (see also Subsection 2.4.6 on Page 38).

Quality assurance

Quality assurance of the reports is carried out by another consultant not involved in the project. This ensures the test results are transparent. Technical editing by SySS also ensures the linguistic and formal quality of the report.

Expert opinions

Findings are always evaluated in the final reports on penetration tests. The experts at SySS are highly trained and have a great deal of experience. In contentious cases they discuss the situation with one another and regularly pass on their acquired wealth of experience to their colleagues.

In order not to affect the neutrality of our reports and expert assessments, the management at SySS has no influence over the expert assessments by the consultants. The consultant always makes an assessment to the best of his/her knowledge and belief. In rare cases the SySS consultants may not find any consensus when evaluating vulnerabilities. This is not based on a lack of expertise or incomplete conclusions, but rather on different approaches. This diversity of opinion is not found solely among the consultants of SySS. Encryption solutions are one example of a typically different assessment by IT security experts. Whereas one group regards strict end-to-end encryption as ideal, the other group attaches more importance to protecting users in the best possible way against harmful content. This approach means that strict end-to-end encryption would be counterproductive.

Tools

The consultant performing a test is responsible for both the exact procedure of a test and the selection of tools. Based on his/her experience, he/she adapts the procedure to the test object and, in particular, to the test depth. He/she also selects the ideal tools. The quality, usability and licensing conditions of software or hardware may change at short notice.

The following overview is therefore also just an exemplary selection of tools which are used regularly to test systems:

- Port scanners such as Nmap, ZMap or scanners developed in-house by SySS are used for system and service detection.
- Nessus, Recon-ng, BloodHound, Metasploit, PowerSploit or the IPv6 attack tool kit can be used as automated vulnerability scanners or analysis frameworks.
- A very large number of tools, e.g. Relayscanner, Smtpmap/Smtpscan, Ike-Scan, Dnswalk or the Hping family, are available for further testing of individual services.
- Manual tests are supported by Telnet, Netcat, Socat, OpenSSL or Stunnel. The Metasploit Framework is also a constant companion in these tests.

Depending on the test object, more specialized tools are used. For web application tests, for instance, SySS uses proxy tools such as Burp Suite Professional and many scanners and systems developed in-house. SySS primarily uses the Aircrack-ng suite to test Wi-Fi infrastructures. Tools such as Hostapd, airgeddon and EAPHammer – at times with their own adaptations – are also used.

The decision as to what tools are actually used is firstly based on the anticipated knowledge gain and secondly on the test depth. Not every tool can be applied in the same way for every software. The use of any tool has a certain minimum run-time; if this time extends well beyond the planned test period, the tool may not be used. If allowed by the planned test period, it is also possible to use tools which were only published, for example, after the start of the test. The planned use of tools can also be discussed with the consultant managing the project before the start of a specific project.

1.2 Limits and risks

A security test is carried out in order to detect and then eliminate many different security vulnerabilities.

1.2.1 Limits of security tests

A security test provides an analysis of the actual state. It is very difficult to detect risks that might arise in future due to possible configuration changes or new findings – these considerations are always speculative. In order to prove their effectiveness, security tests must therefore be carried out at regular intervals. The security gaps which can be discovered also depend on the test perspective.

Another problem is the shortage of budget funds: If large networks or complex web applications are tested during a short period, there is a danger that security gaps cannot simply be identified due to a lack of time. A potential attacker, who takes sufficient time for an in-depth study, can detect and exploit these security gaps.

1.2.2 Distinction of security tests as opposed to other tests

Compared to IT baseline security audits ("IT-Grundschutz") or certifications in accordance with ISO 27001, a security test is primarily concrete and technically oriented – it creates verifiable facts, evaluates the current state of the IT security, and names direct threats.

Security loopholes can also be regularly proven even if a certification is available, regardless of whether it was issued, for example, by the German Federal Office for Information Security (BSI) or a TÜV. Therefore, independent tests carried out by specialists are an important addition.

Security tests and certifications are not in competition: While a certification usually requires compliance with a standard or sets up regulations for information security, a security test shows very specifically what the actual security situation is like – and does so relatively quickly. In addition, the results can be integrated in an existing audit or may even be required as mandatory.

1.2.3 Denial-of-service risk

Great importance is attached to potential denial of service (DoS). First of all, this may occur due to errors in services themselves and also as a result of misconfigurations. Generally speaking, the objective of a security test is not to override systems or applications, but rather to verify whether this is possible and what risk is involved here.

The following procedure has proved highly effective in detecting potential DoS: If a potential denial of service is discovered, we first contact the customer's contact person. During a direct discussion, it is decided whether or not SySS will provide the actual evidence (i.e. trigger the fault). Exploitation of a potential denial of service may be practical if the customer wants a clear indication that changes have to be made to a certain system (maintenance, isolation or even replacement).

SySS does not perform tests whose objective is to impair availability through the use of bandwidth. The risk posed by these attacks always exists and can be determined at any time by analyzing the available bandwidth.

Since a security test is an active control measure, it can never be totally ruled out that the systems to be tested are impaired. Impairments may occur with functions of individual services, the tested service itself and the entire tested system. Especially during testing of web applications, a potential denial of service is not normally assumed. However, these risks also exist here since queries not generated by regular users can be addressed to

the database during a test. It is often difficult to predict these problems. If, however, there are clear indications that these problems exist, the above-mentioned action can be taken.

It is impossible to perform a security test which contains no risks whatsoever. With critical systems, it may be practical to carry out penetration tests on a test system rather than the productive system so that conclusions regarding vulnerabilities in the productive system can be drawn.

In the experience of SySS, two situations are responsible for denial of service during security tests: Firstly, DoS may be triggered by systems or applications which reach their limits even with just a moderate test load. Secondly, very old and neglected services harbor considerable potential DoS. Generally speaking, it should be considered during the preliminary meeting (KICKOFF, see Subsection 1.3.1 on Page 17) whether old or very old systems are to be tested (e.g. with a very outdated patch status) or whether load problems occur anyway in specific systems. In order to deal with the latter problem, agreement can also be reached to perform tests outside peak load periods. Penetration tests only differ sporadically from functional tests, load tests and connection tests. In the same way as these tests, a certain volume of data must be expected in a security test. The systems involved in the security test must process these data just like in normal operation.

The main difference compared with other test methods is that in a security test, various services are confronted with queries which do not occur every day. This is precisely the basic procedure for detecting all kinds of security deficits. It is not possible to deviate from this procedure, unless technical measures are to be dispensed with entirely. Thus, in order to avoid as many potential risks as possible, SySS proceeds as follows:

- Performance of tests by trained and experienced specialists
- Performance of penetration tests only after a written order and clear test approval for the systems to be tested
- Testing of the data supplied by the customer for correctness (e.g. IP ranges); consultation always takes place in the event of uncertainties
- Staging of a kick-off meeting based on a tried and tested method, including preparation of written protocol
- Continuous support in the ongoing project by a contact person from the customer
- Minimization of the risk through design of the test project: However, slow scans (reduction in bandwidth) increase the duration of the test enormously.
- Test can be performed outside of business hours (e.g. at night/at the weekend); if this procedure is required, a contact person for the customer must be available directly at this time.
- Examination of test systems: If the test system only corresponds to the productive system in regard to a few aspects, SySS must always refer to this situation in the final report in order not to distort the significance of the final report.
- Termination of the test if problems are detected: Indirectly created problems are generally not perceptible to an external perspective; therefore, the customer's contact person must be able to clearly assign occurring problems to the test and must, in particular, inform SySS.
- Selection of non-invasive test methods: The resulting reduction in knowledge gained must be accepted; speculation is always denoted as such in the report by SySS.

SySS would like to refer here explicitly to the permanent availability of a contact person of the customer during a test because a number of the above-mentioned aspects will definitely not be fulfilled if this person is not present. In particular, the customer's contact person should also have the skills to plan a different test procedure together with SySS and change, if necessary, any focal points of a test. The organizational processes should make allowance for a certain degree of flexibility. If, for example, the usual contact persons are not available during a test, substitutes must be appointed and vested with the corresponding authorities.

Based on the experience of SySS, the DoS risks can be traced back to four technical causes which are described below.

1. Load during web application tests: During web application tests, in a way similar to function tests, load is generated on the database supplying the web application. This may entail, for example, a search across all fields that appears to be impossible to the user of the application. If the system on which the database is running is very tightly calculated or simply outdated, this may lead to negative effects. Employees of the customer must manually service the database in this case – since only the frequency of the search inquiries can be changed externally, but not the priority of a search compared with others.

SySS recommends that these systems not be designed for a very low estimated utilization volume, but that provision should be made for sufficient performance reserves. These reserves can also be obtained by means of optimizations of the database and the search routine in the web application. However, when old or extremely old systems are used, clear limits are set in regard to load reduction by optimizing the database.

2. E-mails to internal addresses: Corresponding functions can be used to send e-mails, e.g. product or contact queries, via the web application. These functions must not be misused either to send spam mailings or falsify e-mails. Problems also arise here due to systems used for sending e-mails. These systems cannot process a series of automatically generated messages quickly enough. Manual servicing of the participating systems is also necessary here since assumptions of the composition of the participating systems can only be made externally.

3. Failure of infrastructure components: Security tests produce a certain amount of network traffic which must be handled by the participating components, routers and switches. They are expected to function correctly here just like in normal operation. The load produced during a security test also naturally corresponds to the load which would be produced during intensive use of a large number of communication services. Infrastructure components, which fail during these tests instead of becoming slower, must be regarded as extremely critical. Even if the load is higher than during legitimate use, the load in a security test can in no way be compared with that during a distributed attack (DDoS). There is also a risk that the systems cannot absorb naturally occurring load peaks. This may lead, for example, to positive assumption of a new offer by the company's regular customers or reactions by the public to messages. As a rule, these incidents can clearly be traced back to the utilized hardware and/or the software running on the hardware. If this is no longer supported anyway by the vendor, SySS recommends an upgrade or migration. Although the risk can be reduced by a slower procedure during testing, this easily multiplies the time required for the project.

4. Insufficient line capacity: SySS primarily uses root servers in the internet for tests. Both comparable systems and a large number of the utilized tools themselves are generally available. The load required during a security test can therefore also be generated by third parties with comparatively low costs.

In addition to line capacity, the sole decisive factor for the risk is time. A reduction in the required bandwidth is always sold with a longer duration of the test. Alternatively, only random samples are possible. Since the line capacity can only be determined indirectly and imprecisely by an external source, SySS therefore depends on precise and consistent information from its customer. In this case, SySS cannot inspect the contracts or agreements between the customer and his providers, and cannot therefore determine this information itself.

SySS generally recommends that the line capacity be adapted to modern requirements. If, for example, several Class C networks are supplied by a 2 Mbit line, impairments already normally occur due to the insufficient bandwidth. On the other hand, costs can be reduced by externally hosting individual systems either completely or partially. If this is not possible because, for example, lines with corresponding capacity are not available locally at a reasonable price, SySS recommends that a clear concept be formulated in the event that the line is overloaded – provided there is no ill will – and everyday use is then no longer possible. The customer's contact person should be the particular provider in this case.

Ten tips by Sebastian Schreiber

1. Regard penetration tests as a process rather than an individual project. This facilitates and creates a continuous approach which is much more efficient than when the project is viewed separately.
2. Penetration tests are very easy to plan and prepare if they are scheduled at an early stage. Although good companies are able to cope with large projects at short notice, this always leads to avoidable additional expenditure.
3. Be present during the security test! A security test provides you with an unusual change of perspective and is also a special experience to see your own systems "under fire".
4. Whereas some tests have to be performed unannounced out of necessity (e.g. ticket inspections, Economic Control Service, WKD), we recommend that penetration tests be announced beforehand. This creates trust inside the company and strengthens cooperation.
5. Customers are often unsure as to whether to choose a black box test or a white box test. We recommend a happy medium: Specifically provide the penetration tester with the information which he/she requires for his/her project and which a realistic hacker could personally determine anyway (grey box).
6. The neutrality of the tester is his/her most important characteristic. This neutrality is at risk if the tester was involved in creating the test object or the result of a test offers him/her benefits or damages his/her reputation. Security gaps may never be solved using the same mindset with which they were produced (based loosely on Albert Einstein).
7. Often there is no suitable budget or time window to perform a penetration test. However, it would be wrong if the consequence were to dispense with the test altogether. Even a compact test is beneficial for your own IT security.
8. So-called distributed denial-of-service (DDoS) attacks reveal whether or not a system can withstand sabotage attacks. In principle, however, IT resources such as bandwidth or CPU power can also be exhausted without a penetration test. In our opinion, these attacks should be avoided since the anticipated knowledge gain is disproportionate to possible negative impacts.
9. It is obvious that all systems can never be subjected to an exact test. Choosing a representative sample is therefore essential for an efficient test. This is based on practical clustering in which the test objects within a cluster are very similar, but the clusters themselves are different. If a representative from every cluster is tested, a reasonable balance between cost and benefits is obtained.
10. We often see that the follow-up test is constantly postponed in order to eliminate all gaps beforehand. This delays the performance of the follow-up test and paralyzes the continuous improvement process. We therefore recommend that the follow-up test be carried out even if all gaps have not yet been closed, for example four weeks after completion of the main test. A follow-up test normally involves a comparatively small project. Consequently, repeating the follow-up test after another six weeks does not have a significant impact compared with the overall budget.

1.3 Standard test phases

A security test consists of a framework project, standard test phases and individual test modules. Figure 1.1 on the next page shows the project schedule and its design possibilities: This section deals with the standard test phases which normally form integral parts of a penetration test. The detailed description of the test modules, which vary depending on the test object, can be found in Chapter 2.

1.3.1 KICKOFF: Preliminary discussion regarding the project

The project schedule is planned jointly during a preliminary discussion, which normally takes the form of a kick-off meeting over the phone. The consultant conducting the test and a contact person for the customer therefore discuss the following topics, for example:

– Test period and test time window
– Contact persons and their availability
– Discussion concerning the test object
– Necessary preconditions (described in the respective module)
– Handling when DoS potential is detected
– General information regarding implementation (described in the respective module)
– Determination of the language in which the report is to be written (German or English)
– Number of printed copies of the report
– Questions and requests regarding the project schedule

There are individual deviations depending on a special test object and the selected modules. If it is necessary to deviate from the performance of a test as described in the individual modules, this is also discussed at this point in time. The results of the kick-off meeting are recorded and the customer is notified immediately.

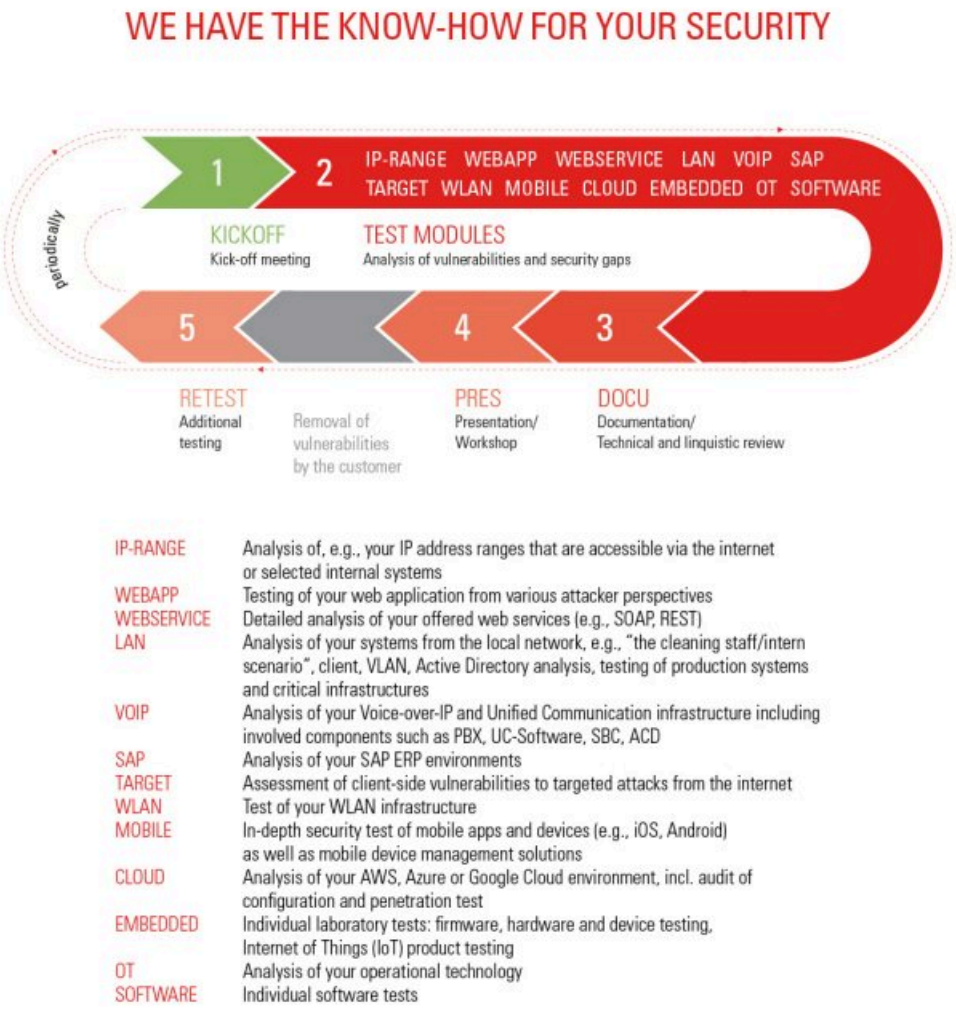

Figure 1.1: Flowchart for a penetration test

Tip by Sebastian Schreiber

Check the points and tips from the relevant module section under "Cooperation by the customer" before the kick-off meeting! The more time you take to prepare and perform the kick-off meeting, the more efficient the test will be and the greater the benefits for your company!

1.3.2 MODULE: Implementation of the security test (selected modules)

The performance of the security test commissioned by the customer is determined by the selected modules. These modules form the centerpiece of every project and embed the test object within its context. The modules provide very general information on what preconditions are required for the security test and how the test will roughly proceed. The modules act as a framework for the security test to be performed and merely provide starting points for the possible procedure because every security test is implemented individually in detail according to the test object stipulated by the customer.

The individual possible test modules are described in detail in Chapter 2 and illustrate the diversity of design possibilities for each security test. Our experience enables us to react even to unusual test requests from our customers. The list of modules is also dynamic because it is extended continuously and adapted to the current requirements, wishes and needs of our customers.

1.3.3 DOCU/REPORT: Documentation and final report

All the test results are summarized in a report at the end of the project. The report includes the following aspects:

- Summary of results ("Executive Summary") and an estimate of the general security level
- List of the detected vulnerabilities with a risk assessment [1] and rectification measures
- Clear presentation of evidence of every detected vulnerability
- Extracts from the printouts of the test tools where this appears practical
- Documentation of special aspects in the test procedure
- If explicitly requested, evidence of communication with the customer

The customer specifies how many printed copies of the report are required. The report is printed out and bound within SySS. It is sent to the customer by registered letter with acknowledgment of receipt (can also be sent outside Germany). The customer also receives the report as a PDF file.

On request, the customer receives the raw data produced during the test (outputs of the test tools). However, this raw data is provided without being processed, for example false positives are not removed from the results of the vulnerability scanners. Unless otherwise agreed, SySS deletes the raw data three months after the end of the test or three months after any follow-up test.

The documentation requirements for internal tests are much higher since, in particular, the test procedure must be described in more detail. The measures for rectifying vulnerabilities are also discussed with the contact person during the documentation process.

The table in the following is an example of a list of security vulnerabilities and proposed measures:

Risk	Findings	Recommendation	Reference
H1.1	**www.kunde.de** Configuration file contains valid login data for the TYPO3 back end	Remove affected data from the server or stop access through ACL	2.1.1 Page 7
H2.1	**198.51.100.1** The router uses a standard password set by the vendor	Change username and password	3.1.1 Page 19

Fortsetzung nächste Seite …

[1] On request, we also offer an assessment according to current scoring schemes such as CVSS or CWSS.

Risk	Findings	Recommendation	Reference
M1.1	**www.kunde.de** Login data and the session cookie are transmitted via an unencrypted connection	Only transmit login data and session cookies via encrypted connections	2.2.2 Page 9
M2.1	**198.51.100.111** The SMB service allows read access to network shares	Filter service on the firewall	3.4.7 Page 29
L1.1	**www.kunde.de** The password guideline enables trivial passwords to be assigned	Revise the password guideline	2.3.9 Page 14
I1.1	**www.kunde.de** The login page can be misused for user enumeration	Do not provide any feedback about the existence of a user	2.3.9 Page 13

DOCU/CSV: Machine-readable finding table

If necessary, SySS can provide the finding table of the final report as machine-readable CSV file in order to process the findings automatically, for example in a ticket system. The CSV file contains only meta information and short information, not the full description, figures or proofs of concept.

On request, other machine-readable formats (e.g. Excel) can be generated with all reports for the same customer. This service can only be provided together with a customer template (TEMPLATE).

DOCU/TEMPLATE: Customer template

Sometimes a customer has further requirements for the final report, for example due to regulatory reasons, so that the typical contents of the final report do not suffice.

In that case, SySS can develop a template according to the customer's needs for all future reports for the customer that allows variations from the default report, such as:

- Customer's project number on the title page
- Remarks for the revision
- Customer's metric
- Additional finding properties for automatic processing
- Modifications to the finding table
- Additional sections
- Uniform structure

1.3.4 PRES: Presentation workshop

The results of the complete test can be shown locally to the customer in the form of a presentation akin to a workshop. A two-part approach has proved its worth:

It starts with a briefing for the decision-making level ("Management Summary") lasting around 30 minutes. Basic results of the test are discussed at a strategic and organizational level during this briefing. On request, Sebastian Schreiber, the Managing Director of SySS, is also available for this part.

The second part is a technical workshop which is aimed at systems managers and administrators. In-depth questions can be asked and potential solutions discussed during this technical workshop. This part is the responsibility of the consultant in charge of the test.

1.3.5 RETEST: Follow-up test

The purpose of a follow-up test is to measure the effectiveness of the measures for rectifying security vulnerabilities which were detected in previous tests.

Once the security gaps have been identified in a test, they must be rectified. Although no special consulting services are normally required here, the results of these rectification measures should be verified. It is therefore advisable to carry out a follow-up test, for example after two to four weeks, but at the latest after six months. A search for new vulnerabilities is not performed during a follow-up test. Instead, the status of the already known security gaps is examined and documented. The procedure is discussed beforehand. The report already produced on the main test is used as the basis for the documentation and the "Executive Summary" is adapted. An estimate of the time and effort required for a follow-up test is only worthwhile if the results from the main test are known.

1.4 Penetration tests in agile environments

The findings of a penetration test always refer to the specific time of the analysis. For example, if the test is repeated one year later, the situation encountered is always different, which is primarily due to the following two points:

– The tools for attacks/hacking methods have become more powerful and/or more efficient.
– The test object itself has changed.

Alongside regular penetration tests, penetration tests due to events are also of great importance, especially for agile developments such as applications, mobile apps, etc. In order to cope with changes to the test object, a penetration test is often performed before every release. As a result, the penetration test represents a "quality gate" that must be passed before a release goes live. In some cases, a successful penetration test is even an acceptance criterion agreed in a contract for work.

New releases are created very frequently with agile developments. If penetration tests are increased by the same extent, the costs for pentesting also multiply with the increase in releases – and there is a danger that safety will fail due to the cost factor. In order to prevent this eventuality, the following alternatives to very frequent penetration tests are available:

– Avoidance of situational penetration tests for agile developments
– Testing of e.g. only every third release or every release that is approved by the product owner
– Differential penetration tests: testing only of components that have changed since the last penetration test

The third alternative appears attractive. The prerequisite for this, however, is that it can be ascertained which paths in a web application must be visited in order to test these websites and forms whose security may be affected by a concrete source text change. If, for example, the new release of an online shop contains a payment interface for the first time, this is the only part that would be subject to the penetration test. Nevertheless, if the efficiency of an application or the interface to a database are optimized or a large number of small changes are programmed, this can affect the security of all the application components: another full test would be performed. All three alternatives appear to have disadvantages and initially convey the impression that it would not be possible to perform a good penetration test in agile environments. However, this is a misconception. It is based on the idea that vulnerabilities are detected "only" in order that they may be fixed. We recommend a broader perspective, even for developments: The developers' internal processes and mindsets must also be optimized in parallel with improvements to vulnerabilities because the next development is never far off! Simply marking off the findings of the penetration test in the list of vulnerabilities with "fixed" misses a key benefit. It is important to get the root of the issue, which means using penetration tests to aid the development in a sustainable manner – and it is precisely this benefit that is completely independent of the frequency of releases.

Tip by Sebastian Schreiber

Use the penetration tests that have been performed in agile environments not only purely to eliminate vulnerabilities, but also and more importantly to test the development of your product. It is therefore worth asking and finding answers to the following key questions:

- What did we do wrong which meant that our product had vulnerability XY?
- What did we not take into account when planning?
- How can we improve and ensure that this vulnerability does not occur again in the future?

2 Penetration Tests – Test Modules

This chapter contains detailed descriptions of some of our standard test modules. If you also request a check on a test object which cannot be covered by one of our standard test modules, an individualized test procedure is formulated and proposed.

2.1 IP-RANGE: Analysis of selected systems

Summary

Selected internal IP addresses or IP addresses accessible via the internet, or also whole IP address areas are checked for concrete security vulnerabilities and the resultant risks are evaluated.

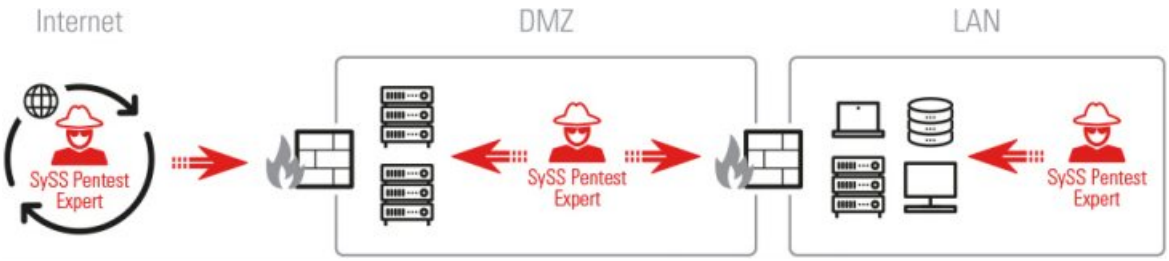

Figure 2.1: Possible forms of the module IP-RANGE

Starting situation

Within the framework of this module, individual systems or groups (clusters) of systems are analyzed in regard to vulnerabilities. This may firstly involve selected internal systems, e.g. the infrastructure of an important application environment, or analysis of "critical infrastructures" such as the network segment for industrial control systems and SCADA systems. Secondly, however, the analysis may also include systems which are directly accessible from the internet, thus making them highly exposed to risks, and are operated, for example, within their own demilitarized zone (DMZ).

There are several types of risk, especially during the operation of systems via the internet. For example, there are security vulnerabilities in individual services that allow third parties the following opportunities:

- You may be able to obtain detailed information about the systems and the utilized software which are useful for other attacks.
- You can examine confidential data and information which do not relate to the system or software.
- You can penetrate the system and use it for your own purposes or further attacks.
- You can manipulate data.
- The availability of systems can also be restricted.

Objective

The objective of a security test within the framework of the IP-RANGE module is to check the systems to be tested, depending on the test depth for the above-mentioned risks. As described in the "Denial-of-service risk" section (see Subsection 1.2.3 on Page 13), the objective of a test is not to stop systems or services, but merely to discover the potential denial of service.

Web applications or web services are not tested within the framework of the IP-RANGE module. The WEBAPP (see Section 2.2 on the next page) or WEBSERVICE (see Section 2.3 on Page 29) modules are used for this purpose. An impairment analysis of the detected data is not carried out since past experience shows that our customers can and must also easily do this themselves.

Implementation

The implementation method is defined by the consultant in charge, but normally follows this pattern:

- Verification of the correctness of the data provided by the customer
- Identification of operating systems and accessible services
- Testing of the recognized services using vulnerability scanners
- Verification of the results, verification of recognized security gaps
- Use of tools which cover areas not considered by vulnerability scanners
- Manual tests
- Proof of potential DoS in agreement with the customer

Using the available information as a basis, the consultants conducting the test select the tools which best ensure the success of the test. Some examples can be found under "Tools" (see Subsection 1.1.6 on Page 12).

The pentest box developed by SySS (see Subsection 2.4.6 on Page 38) can perform the IP range test, the objective of which is to analyze internal systems or systems in a DMZ, which can also be done remotely.

Cooperation by the customer

Some general conditions are needed to ensure that a security test can be carried out efficiently and with substantial benefits for the customer. Otherwise, the test becomes more difficult or delays and additional costs are incurred.

- The IP addresses of the systems to be tested must be available in good time before the test actually starts.
- Written consent must be obtained from third parties when their systems are tested.
- Maintenance windows, time zone dependencies (when testing systems outside Germany) and public holidays must be observed when selecting the test period (during the kick-off meeting as described in Subsection 1.3.1 on Page 17).
- Contact persons should actually be available and capable of taking action during the above-mentioned test period.
- The contact persons should have an overview of the internal responsibilities for the tested systems as this reduces the communication effort during the test.
- Our source IP addresses should be temporarily activated in protective mechanisms that may be used, e.g. intrusion prevention systems.
- The responsibilities should be clarified.

You can further increase the quality of the test if you have your documentation on the IP addresses to be tested checked prior to the test.

Tip by Sebastian Schreiber

Before the start of the test, make sure that the affected employees and systems managers are informed so that the test is positively perceived.

2.2 WEBAPP: Testing of web applications

Summary

Selected web applications are tested for their security from different perspectives. A search is made here for security gaps which are based on the utilized software, its configuration or the application logic. Underlying systems (providing infrastructure), e.g. web servers, application servers or database servers, are also tested for vulnerabilities.

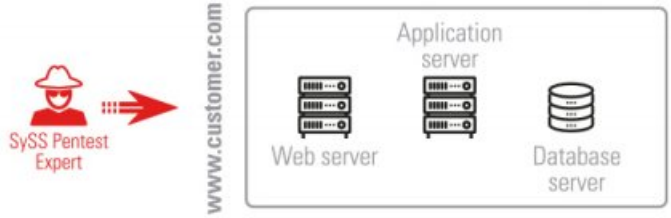

Figure 2.2: Module WEBAPP

Starting situation

Generally speaking, web applications pose a high risk since customer or third-party data can be accessed without authorization, a fact which is tantamount to a loss of confidentiality. The interaction between the user and the website is also relevant since both external and internal users may be at risk, especially on account of cross-site scripting (XSS) vulnerabilities.

One special aspect of web applications is the often complex dependence on other systems, e.g. application servers or database servers. They may also be impaired by vulnerabilities in the application. In the case of a database connection, even sensitive information could possibly be obtained from the database. These dependencies may also exist in regard to utilized middleware – and in organizational terms – commissioned suppliers. The security of a web application cannot be solely defined by it itself, but also by a content management system (CMS) that is possibly used as a basis. Another special aspect of web applications is that many security problems can also be understood by laypersons after they have become known. This is often accompanied by a negative image and a loss of confidence.

Objective

The objective of this test module is also to determine whether the above-mentioned risks exist. Assessment of the risk of an individual vulnerability is very important in this test since the existence of a specific risk is a clear indication of a general problem in regard to application development: for example, an XSS gap indicates generally poorly implemented server-side input validation. Another focal point in the test deals with the question of whether it is possible to inspect customer or third-party data by exploiting typical vulnerabilities of web applications. The security level of the application is finally estimated and measures to remedy any vulnerabilities are proposed.

Implementation

Implementation of a penetration test with web applications depends a great deal on their function and configuration. Although a fixed scheme for the test procedure cannot therefore be established, the procedure roughly corresponds to the following pattern:

Testing of the providing infrastructure: If only a web application test is commissioned, but not a thorough analysis of the providing infrastructure, other services possibly available on the web server will also not be tested. If an analysis of the providing infrastructure is explicitly requested, port scans are also carried out to determine other services which are accessible on these systems. All services are then subjected to a vulnerability analysis (as in the IP-RANGE module, see Section 2.1 on Page 23). For example, weaknesses in the configuration of the utilized web server itself and in the utilized SSL/TLS components are also analyzed. Depending on the accessible services, different tools are used in this case. Vulnerability scanners such as Nessus are part of the standard repertoire in these tests. The objective of this check is to also identify vulnerabilities outside the services provided for the web application.

Structural identification: With regard to the actual test of the web application, the structure of the web application to be tested is analyzed during an initial phase. In addition to different automated methods (spider/crawler), manual methods are also used in this case. The prime objective of this phase is to identify the application areas which are of interest to an attacker.

Test of the authentication concept and session administration: If the application requires user login, the next step is to determine possible attacks against the authentication concept. In addition to purely technical vulnerabilities (e.g. circumvention of authentication by SQL injection attacks), programmatic conceptional vulnerabilities (e.g. possibilities of user enumeration, password reset functions, password-guessing attacks, account blocking, etc.) are also evaluated in this respect. All other tests are ideally performed using several user accounts with – if possible – different roles. The implemented session concept – if available – is then examined. This focuses on basic vulnerabilities which may make it possible to steal identities. Session IDs, cookie attributes, session handling and pre-authentication vulnerabilities are tested, for instance.

Testing of input validation: The focal point here is testing of the functionality of server-side payload verification by means of classic attack vectors (different forms of cross-site scripting, SQL injection, URL injection, LDAP injection, OS command injection, XPath injection, XML injection, etc.). In addition to manual input tests, payload manipulations are carried out for this purpose using browser plug-ins and analysis proxies, e.g. Burp Suite Professional. Where appropriate due to the extent and complexity of the application, automated web application scanners are used. The results from these scanners can be verified manually afterwards.

Analysis of application logic and the authorization concept: In another step, the application is analyzed in regard to potentially erroneous consistency tests or plausibility tests within the application logic. Classic examples here would include the possibility of price manipulation in an online shop, susceptibilities to fake replies by integrated third-party systems – e.g. payment service providers – or unauthorized branches within the application logic, e.g. through manipulation of logic components moved to the client which are transported, for instance, via hidden fields. The authorization concept implemented by the application is also tested. In an ideal scenario, several test accounts should also be provided for this purpose in order to be able to efficiently test possibilities to access data and functions of other users or roles.

Reverse engineering: Optionally, client components supplied by the server (also binary), e.g. Java applets or flash applications, can be analyzed. Special decompilers and reverse engineering techniques are used for this purpose. Depending on each finding, an attack software suitable as a proof of concept (PoC) is developed in principle. The main objective here is to verify the findings and acquire additional information or privileges.

OWASP top 10: When looking for vulnerabilities, SySS strongly aligns itself with the current OWASP top 10. For many years, the Open Web Application Security Project (OWASP) has been keeping a list of the ten most common risks for web applications. This dynamic list is revised at regular intervals. In addition, a number of

other vulnerability categories are tested although they are not contained in the OWASP top 10 or have only recently been newly published.

Vulnerability scanners only play a secondary role in this test module. This also applies to those vulnerability scanners which are specially intended for testing of web applications since scanners can only use and evaluate context-related information to a very limited extent. The main tool when testing web applications is therefore always an internet browser with which manual tests are performed. Mozilla Firefox is used preferably in this case since a large selection of add-ons are available for this browser. If required, own scripts are also written to verify and demonstrate vulnerabilities. However, testing of web applications can also be meaningfully supported by using security scanners or corresponding proxies. Nessus, SQLmap and Burp Suite Professional are used here, for example.

Cooperation by the customer

The web application URL must always be notified for the test. It must also be defined what areas of the web application are to be tested. Examples of this include publicly accessible areas of the page, functions only available to registered users and any areas belonging to a possibly used content management system, e.g. administration interfaces. It must also be clarified from where the test is to be performed (via the internet, from a specific intranet area or similar).

Contact person: An analysis of web applications does not only differ significantly from other test modules in terms of the procedure. As already mentioned, security problems in the web application can also affect other services, especially databases and e-mail services.

Web applications and/or their functionality also do not normally come from a single source: Their design may be the responsibility of an agency while programming of the web application may be carried out by both internal and external programmers. The hardware itself can in turn be provided and also supported by a web hoster.

In particular, in order to rectify the determined security vulnerabilities, it is necessary to contact those persons who are responsible for each affected element. It is therefore extremely important for the consultant performing the test to know the contact details of the contact persons.

If a direct contact person has not been appointed or is unavailable, this may cause two kinds of problems:

– Queries during the test, which are used to verify security vulnerabilities, cannot be answered. This subsequently leads to delays.
– If the persons responsible for a function affected by a security gap are not known, rectification of the vulnerability is delayed considerably.

Clarification of the competencies and appointment of the responsible persons should therefore start in good time before the test, even if this initially appears time-consuming. The affected persons should then be informed about the date and objective of the test. Just like in other test modules, these persons can be present during the test if they so request. If third parties are affected, they must agree to the test being performed (in the form of a written declaration of consent). A contact person familiar with the web application from a user perspective should also be present during the test to be available for queries.

The tests are always carried out in close cooperation with the contact persons. This firstly ensures that any availability problems which occur can be detected immediately and then rectified. Secondly, it is ensured that critical vulnerabilities, in particular, can be rectified immediately.

Dependencies: Organizational and technical dependencies should be notified to SySS. This can take place during the kick-off meeting.

Status of the web applications: The functions to be tested must be available the whole time if possible. In the very early or middle phase of implementation of a new web application, a test does not produce any lasting results. However, a test may still be beneficial if decision-related results have to be produced as soon as possible.

Furthermore, no updates should be performed during the test period. The reason for this is firstly that the test depth can be reduced, since during the update no tests can be performed. Secondly, no reliable statements can be made about the current security status of the application if functions change during the test period and therefore cannot be checked comprehensively.

Login information: A large number of web applications offer additional functions in an internal area which only becomes visible after a successful login, e.g. at a customer portal.

In order to test these functions, SySS requires at least two user accounts with different privilege levels based on which the test is to be carried out. If no corresponding accounts are available, a test can only be performed from the perspective of a user of the website. This often leads to only a few findings regarding the security level of the web application. Since experience shows that an attacker who has sufficient time and uses social engineering techniques is most probably in a position to take over a user, it is recommended to provide the test user with this, primarily because the penetration test takes place within a time-limited framework. In order to obtain meaningful results, the privileges of user accounts compared with those of regular users may not be restricted. The way in which these accounts are generated is discussed during the kick-off meeting (see Subsection 1.3.1 on Page 17).

Test data or test system: If the test is not to be carried out with productive data or on productive systems, work can also be performed on a productive system using test data or only on one test system. Test datasets, which can be accessed by the user accounts utilized for the test, should already be available before the start of the test.

When working with pure test systems, the result of the security test is only significant if its functionality largely concurs with that of the productive system.

If a test instance cannot be provided, the procedure is adjusted accordingly. This will ensure, for example, that users of a productive web application to be tested are not exposed to availability restrictions during the test. In particular, the automated vulnerability scans are very moderately configured in such a case or are only used with selected functions.

Tips by Sebastian Schreiber

Elimination of weaknesses detected in a web application requires willingness to cooperate from the person responsible for the affected element. Therefore, try to find out and inform all responsible and affected persons at an early stage – this is the only way to ensure a quick reaction.

Inform all the participants, including those persons who do not have any active tasks during the test itself. You therefore create additional confidence in the security tests as a service and strengthen the position of IT security inside the company. If we do not have any login information for the test, we can often only perform a less meaningful test.

2.3 WEBSERVICE: Examination of interfaces (APIs)

Summary

Selected web services are analyzed from different test perspectives for security vulnerabilities which enable an attacker to endanger the confidentiality and integrity of data or interrupt the availability of the provided web service functionality.

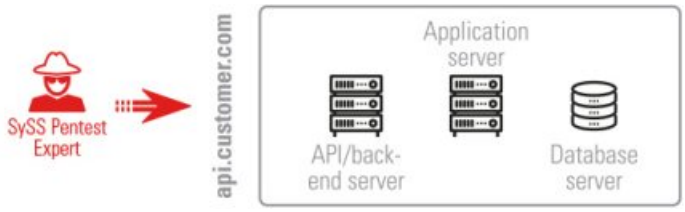

Figure 2.3: Module WEBSERVICE

Starting situation

Just like in classic web applications (see WEBAPP module, Section 2.2 on Page 25), there is also a risk with web services that data can be accessed without authorization, which means a loss of confidentiality. There is also a risk that data can be manipulated in an unauthorized way, thus infringing their integrity. Interference with the availability of web services (denial of service) often also represents a high risk since critical business processes no longer function properly without corresponding web service functionality.

Web services rely on different web technologies and protocols which are also used for the most part in classic web applications. Special formats or a special syntax within customary HTTP queries are normally used in this respect (e.g. in the form of REST API or XML-based SOAP). In addition to typical security vulnerabilities in back-end applications such as buffer overflow, SQL injection vulnerabilities or (de)serialization, there are also web service-specific vulnerabilities such as XML/XPath injection or XML signature wrapping which may pose a security risk to web services.

Objective

During the security test, the selected web service is checked for vulnerabilities from different perspectives. Depending on the focal point of the test and the functionality of the web service, e.g. in the context of business-to-consumer (B2C) or business-to-business (B2B) transactions, the emphasis may be on different protection objectives (e.g. confidentiality, availability, integrity). Furthermore, for every security analysis of a mobile app[1] the corresponding web service is tested. SySS does not perform load-based denial-of-service attacks; however, checks are made as to where DoS may occur due to misconfiguration of the server or erroneous implementation of the web service.

Any utilized authentication protocols such as OpenID Connect (OIDC) or OAuth and single sign-on services such as Active Directory Federation Services (ADFS) and their specific implementation are also tested.

[1] For more information on analysis of mobile apps, see Subsection 2.9.2 on Page 54.

Implementation

Performance of a security test of web services is primarily based on the utilized technologies and architectures (e.g. SOAP, RESTful, JSON-based, etc.) and the supplied functionality which is documented in the web service specification.

The test procedure is normally based on the following pattern:

- Analysis of the web service specification
- Threat analysis to identify possible attacks, e.g. regarding unauthorized attacks on external data or unauthorized manipulation of external data
- Checking of vulnerabilities in back-end systems (e.g. buffer overflow, SQL injection, XML injection, (de)serialization)
- Checking of further vulnerabilities in web services (e.g. XML/XPath injection, XML signature wrapping)
- Verification of access control/session administration (if available)
- Search for errors in the application logic of provided functions

Both special software tools, such as SoapUI, Burp Suite Professional or Postman, and manual test methods are used to carry out the security analysis.

Cooperation by the customer

A web service's URL as well as its specification or interface description, e.g. as a WSDL or OpenAPI file, must be provided for the security test. Examples for inquiries should also be provided each time and the possible authentication described. This is important if authentication of an identity provider is carried out, but which is not part of the test. Past experience shows that the description of the interfaces is often not sufficient to guarantee trouble-free communication with the service.

The security analysis of web services generally corresponds to that for web applications. This is the reason why the same organizational and technical aspects must be considered (see WEBAPP module, Section 2.2 on Page 25).

Contact person: Responsibilities should be clarified and the responsible persons determined in good time before the start of a planned security test. Everyone involved in the project should also be informed in advance about the date and objective of the test. A contact person for queries regarding the web services to be tested should also be available during the test period.

Dependencies: Organizational and technical dependencies of the web service should be notified to SySS, including for example the forwarding of data to other services. This can take place during the kick-off meeting. If the web service or downstream services are provided by third parties, SySS requires a written test authorization before the start of the test.

Login information: If access control should be implemented for use of the web service to be tested, SySS requires at least two user accounts per privilege level, from the perspective of which the security test is to be performed. If no corresponding login data is available, a test can only be carried out from the perspective of an external attacker without valid login data. In view of this limited test perspective, security vulnerabilities cannot normally be found in an authenticated context of a web service. Furthermore, there should also be a description of how authentication is carried out. If authentication, for example, is carried out by an identity provider which is not part of test, this may not appear in the documentation of the web service.

Test data/test system: Just like in security tests of web applications, it must be possible when analyzing web services to examine both productive systems using corresponding payload and pure test systems using test data within the framework of a security analysis. Test datasets, which can be accessed by the user accounts utilized for the security test, should be available before the start of the test.

When working with pure test systems, it should also be noted that results of the security test are only meaningful and can be transferred to the productive system if their functionality and architecture are largely identical.

Tip by Sebastian Schreiber

During such a project, always provide the consultant, if possible, with all the available documentation on the interfaces to be tested (including a description of the interfaces and example inquiries) – this saves valuable time during the test!

2.4 LAN: Security test in the internal network

Summary

Different attack scenarios in local networks are described and the risks which they pose are evaluated. Without any customer-specific stipulations, the main objective is to determine possible privilege escalation paths. Although there are generally a large number of opportunities for internal security analyses, some continuously requested test scenarios have emerged over the years. SySS therefore offers specifically described test modules for these test scenarios. In principle, however, there are no limits to the test possibilities and SySS also always finds the right approach for individual concerns.

Starting situation

Unlike systems in the internet that are exposed to the risks from a non-restrictive user group, a security test in an internal network (company network) examines the risk posed by internal attackers. In concrete terms, this means a user with access to the internal network. Due to his/her position, this user automatically has a higher level of knowledge of the network. This may not necessarily involve a company employee with evil intentions because visitors to a building can also potentially attack the company network. Attacks can also be carried out via compromised access data or via malware-infected systems of the company's employees.

Depending on what test scenario is to be evaluated, the tests are initiated from different perspectives. One starting position, for example, is that the SySS consultant is only provided with physical access to the network, but no further information beyond that (see LAN/CLEAN module, Subsection 2.4.1 on Page 34). Another starting position is that the consultant assumes the role of an "intern" (see LAN/TRAINEE module, Subsection 2.4.2 on Page 34). However, targeted analyses of certain application environments or utilized technologies can be carried out. Examples of these are the VOIP-UC and SAP modules, see Chapters 2.5 and 2.6. From module LAN/CLEAN, the different classic test modules of SySS are described in detail.

However, the following section firstly explains some general aspects which must be observed when performing a LAN test.

Objective

Depending on the adopted perspective and the implemented test module, the objectives are to detect and evaluate any existing risks in the company network and put forward proposals for eliminating these risks. The test not only evaluates pure security gaps, but also configurations or the availability of certain software which might provide an internal attacker with some approaches for a successful attack. Not only is the vulnerability of individual systems estimated in this case, but also the communication between services in order to determine

machine-in-the-middle susceptibilities or other protocol-based attack potential. If other protection measures (update, configuration change, replacement) do not prove to be effective, it is normally recommended that the affected systems be isolated internally.

SySS does not perform an organizational file access authorization test in this respect. These checks often cannot be performed by external parties.

Implementation

In principle, implementation for many test scenarios of the module LAN is similar to that of the IP-RANGE module, where provision is made for network-based analysis of certain systems:

– Testing of the systems for accessible services
– Testing of the services using automatic vulnerability scanners
– Verification of the results
– Accompanying and manual tests

There are also some test aspects which can only be realistically implemented in internal networks, e.g. performance of machine-in-the-middle attacks or attacks which require physical access to a system.

Concrete implementation of classic SySS internal test scenarios are outlined from Subsection 2.4.1 on Page 34.

Cooperation by the customer

Logistical preconditions must be fulfilled for an internal test since such a test does not represent a stand-alone service.

Selection of samples: Due to the enormous number of testable services that are usually available in internal networks, great importance is attached to the selection of practical samples. It is imperative that this is taken into account in the kick-off meeting (see KICKOFF, Subsection 1.3.1 on Page 17).

When selecting samples, it should be ensured that the systems are representative of further systems. Ideally, integration systems or test systems are examined in greater depth – especially if the productive systems are required for critical tasks.

With very large or extremely complex internal networks, SySS can help to select suitable samples and also, if necessary, carry out an internal inventory (see RECON module, Subsection 2.14.1 on Page 77).

Contact person: A contact person should be easily contactable in the short term throughout the entire test period. He/she is welcome to attend the test. Since the test normally takes place locally, all general organizational conditions should have been fulfilled before the start of the test.

Provision of information to the participants: All systems managers, administrators and other affected employees should be informed about the test and its objective before the start of the project. Their cooperation may be necessary during the test, but it is very important especially to eliminate any detected vulnerabilities. It should also be checked whether it makes sense to include the company IT security or employee representation in preparing for the test.

Workstation: A workstation should be available for every consultant. The following are required to test internal network components:

- At least one network connection (Ethernet) from which the network components to be tested are accessible
- Power connection for a notebook and switch (socket strip)
- Space for approx. two notebooks, a switch and documents
- Quietest possible environment
- Internet access for documentation and, if necessary, research
- Depending on the test module, a reference device is also required (e.g. a standard client such as a desktop PC, notebook or a thin client)
- At least one user account is also required for some scenarios (e.g. Active Directory user with standard privileges)

Access to buildings and the network: The consultant should be able to enter the affected building with his/her equipment on the day of the test and reach and set up the above-mentioned workstation. Any necessary licenses should therefore be obtained in good time.

If a network access control mechanism is also to be used (e.g. 802.1X with client certificates), corresponding access data should be prepared for the consultant and handed over to him/her at the start of the test.

Handling unstable systems and legacy systems: In internal networks, frequent use is made of systems which are not particularly resistant to attacks and are operated for durations that extend far beyond their end-of-life (EOL) announced by the retailer. The risks that these systems will crash during the test cannot be ignored. This is particularly relevant for production-related machines. Close coordination with the contact person is therefore necessary when testing these systems. Ideally, SySS is supplied with a list of these systems before the start of the test.

In general, SySS recommends testing only systems which have reached their EOL or have no longer been updated for several years if it is to be proved that systems have to be isolated or replaced and the incurred damage can be accepted by the customer. If possible, systems which do not fulfill any critical functions should also be selected for these tests. Liability for any damage resulting from crashes or other impairments is excluded according to our General Terms and Conditions (GTC).

Tips by Sebastian Schreiber

You can improve the quality of the results of an internal test by carefully selecting the systems to be tested.

In many identical systems, it nearly always makes more sense to carry out a more in-depth test of two systems selected as samples than a rough test of all systems.

The objective of a security test is to uncover technical deficits. You should therefore inform all participants before the start of the test so that it is perceived as a positive measure which improves the company's security.

2.4.1 LAN/CLEAN: Cleaning staff scenario

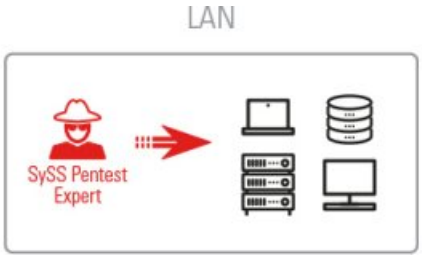

Figure 2.4: Module LAN/CLEAN

This scenario simulates an attack from people external to the company who have had the opportunity to access the customer's network, which may be the case via patched network sockets in public areas or via external devices that are brought into and connected to the company network. The consultant will use his/her own notebook to analyze the security of the customer's network. He/she will look here, in particular, for obvious and easy ways to exploit vulnerabilities ("low-hanging fruits"). The typical procedure for such a test is as follows:

- Testing of any existing network access controls
- Determination of utilized internal network areas
- Identification of active systems and services
- Vulnerability analysis and exploitation
- Privilege escalation and distribution

The objective of this procedure is to obtain the most accurate overview possible of the security level of the internal network environment. Tests can be carried out both in terms of width – i.e. no restriction is placed on the systems to be tested – and depth. In the latter procedure, the customer compiles a meaningful and representative list of the systems which are to be tested and on which the consultant then concentrates.

Depending on the customer's request, the consultant will show ways in which he/she can, for example, extend his/her privileges within the company network, access confidential data or manage specifically to compromise certain systems.

Tip by Sebastian Schreiber

If you inform the consultant about the internally used network areas – in the form of a network plan, for example – at the start of the project, it is possible to save valuable time which can then be invested in the actual vulnerability analysis.

2.4.2 LAN/TRAINEE: Trainee scenario

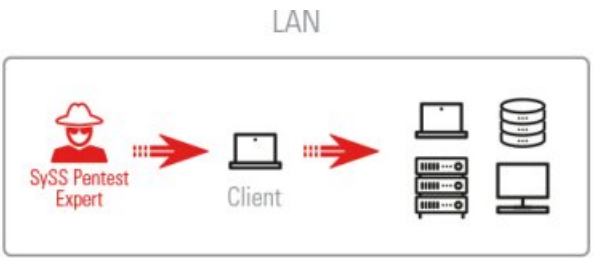

Figure 2.5: Module LAN/TRAINEE

In this scenario, SySS simulates an attack which is carried out using the employee or trainee rights on the customer's internal network. This requires the consultant to have a standard client and a user identification with standard rights. An attempt is then made, from this simulated perspective, to extend their own privileges locally on the client as well as within the network.

The core elements of this test include the following, for example:

– Physical attack possibilities such as booting of external media
– Software inventory and determination of the patch status
– Configuration analysis
– Testing for hardening measures
– Local file system analysis (e.g. NTFS access privileges)
– Inspection of network drives and shares

Tips by Sebastian Schreiber

Present our consultant with the most realistic client possible that would also be provided to a new employee, e.g. a trainee. Also make sure to apply for a test user account at an early stage so that it is available to the consultant at the start of the test. The user account should be equipped with typical privileges.

2.4.3 LAN/CLIENT or LAN/SERVER: Hardening analysis of a client or a server

Figure 2.6: Module LAN/CLIENT or LAN/SERVER

LAN/CLIENT

This scenario simulates the situation where an end device ends up in the hands of someone external to the company (e.g. if a laptop goes missing). Various states (laptop switched off, switched on but locked or unlocked) are used as starting points. By contrast with the trainee scenario described in Section 2.4.2, a careful and extensive hardening analysis of a client is carried out in the course of the module LAN/CLIENT. This analysis examines all attack aspects including attacks against any hard drive encryption solution and its pre-boot authentication. Other test aspects are:

– Virtualization of the image, memory analysis, etc.
– Boot- and hardware-based attacks (booting of external media, PXE, direct memory access-based attacks)
– System analysis (access to confidential data, data loss scenarios, device control, access privileges, malware susceptibility/trojanization, configuration)
– Privilege escalation (on-board resources, vulnerabilities in the operating system and software, exploits)
– Analysis of third-party software (anti-virus solution, endpoint protection, software distribution, etc.)
– Network-based analysis (port and security scans, manual testing, traffic analysis, penetration of the company network, e.g. via VPN)

LAN/SERVER

With the LAN/SERVER module, we also offer a sound hardening analysis of a server image or a server reference installation. The following aspects, for example, are examined in this case:

- Service configuration (in particular, the configuration of network services is evaluated from the perspective of security; examples: web servers such as Apache or Nginx, application servers such as Tomcat or WildFly, SSH, MySQL, MSSQL, SNMP, third-party supplier agents, and much more besides)
- Privilege analysis (Which users have effective privileges on the server?)
- Privilege escalation (on-board resources, vulnerabilities in the operating system and software, exploits)
- Analysis of third-party software (anti-virus solution, endpoint protection, software distribution, etc.)
- Network-based analysis (port and security scans, manual testing, traffic analysis)
- Testing compliance with IT security requirements or best practice recommendations of organizations such as the BSI or NIST if required

Implementation of the LAN/CLIENT module or the LAN/SERVER module is recommended, for example, during a forthcoming migration from an older to a newer operating system (e.g. replacement of Windows 7 by Windows 10 or Red Hat Enterprise Linux 7.x by Red Hat Enterprise Linux 8.x). Before the actual rollout, our consultant will determine during this test whether the implemented hardening measures are effective and whether additional protective mechanisms should be implemented.

Tip by Sebastian Schreiber

The ideal time to carry out a thorough client or server analysis is **before** the blanket rollout or **before** productive deployment. However, thanks to opportunities which are available, for example, in the form of group guidelines, identified vulnerabilities can also be removed afterwards.

2.4.4 LAN/AD: Security analysis of the Active Directory environment

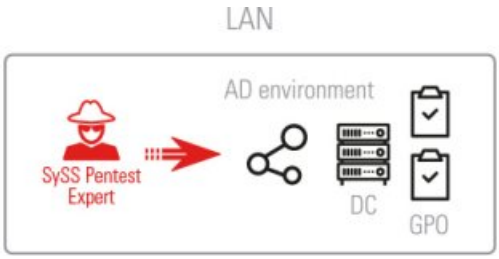

Figure 2.7: Module LAN/AD

The Active Directory (AD) service is almost omnipresent and is used in nearly every company to administer the IT infrastructure. The Active Directory cannot only centrally administer users, groups and computers in the form of object containers – configurations and security settings can also now be controlled in many variations using Active Directory resources in the form of group guidelines. Attackers are also aware of these advantages. In particular, these features offered by the Active Directory are often the reason which actually enable an attacker to reach far into the network or enable a privilege escalation. Great importance should therefore be attached to protection of this extremely important service and use of the large number of offered security functionalities when it is necessary to protect a company's IT infrastructure.

SySS therefore offers dedicated security analyses of an Active Directory environment of the customer. The following test steps are normally the focal points of these analyses:

– Determination of the Active Directory structure (e.g. sites, forests, domains, subdomains, OUs, etc.)
– Identification of the positions of trust between different subareas of the Active Directory environment (e.g. external trusts, forest trusts, cross-links, etc.)
– Analysis of the security-related configuration options (e.g. inspection of the already implemented group guidelines and recommendations for additional security-related group guidelines)
– Evaluation of (different) password guidelines
– Least privilege analysis (calculation of the number of "critical" accounts such as obvious and hidden local domain administrators or domain administrators on critical systems through recursive group dissolution)
– Analysis of any connection to or networking with Microsoft Azure AD

This analysis is normally carried out in close cooperation with the customer's contact persons. For example, a configuration is ideally examined by means of a joint walk through the different group guidelines. However, certain subaspects can also be implemented independently by the consultant. For example, the consultant can use on-board resources such as the PowerShell to obtain a great deal of information about the Active Directory environment.

The objective of this security analysis is to help the customer make his Active Directory environment even harder. In particular, the recommendations by SySS aim to make it much more difficult for an attacker, who has already successfully penetrated the network, to broaden his attack. This can mainly be achieved already with Active Directory on-board resources.

This special test module can ideally be combined with the two classic internal attacker scenarios (LAN/CLEAN and LAN/TRAINEE modules) since they provide the consultant with almost all technical preconditions.

Tip by Sebastian Schreiber

Provide the consultant – at the least when requested – with suitable "interviewees"! Specific questions can therefore be answered directly by each responsible contact person. The particular contact person can also provide the consultant with "proof" of certain security settings by jointly inspecting the configuration without the consultant requiring a highly privileged (test) user account.

2.4.5 LAN/VLAN: VLAN analysis

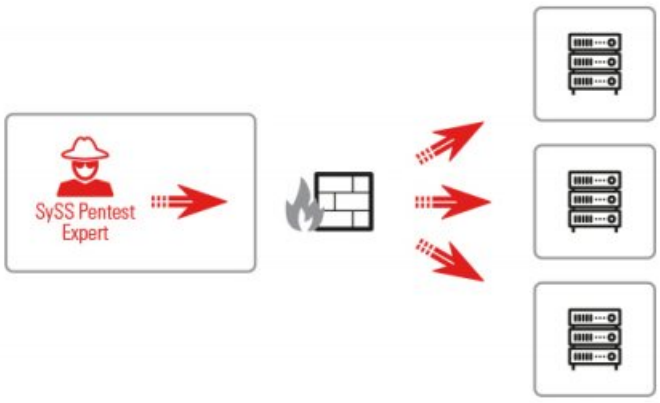

Figure 2.8: Module LAN/VLAN

Device groups such as servers, clients, VoIP systems, printers, etc., are often split into their own networks. These network separations generally occur logically using dedicated VLANs.

In addition to organizational advantages, a network separation also offers the opportunity to check and regulate cross-network data traffic at a central point. In some cases, the separation is supplemented with a network access control so that only legitimate systems are allowed access to the company network.

If switches, packet filters and access control systems have configuration vulnerabilities, there is the danger of an unauthorized incorporation of an attacker into the company network and extension into sensitive and critical network areas (e.g. server network).

For the LAN/VLAN module, the risk of an attacker with the opportunity to access a network connection in the company network is assumed.

In the test, SySS primarily examines a possible circumvention of the access control system and cross-network communication options.

In particular, the following aspects are considered, for example, during the VLAN analysis:

– Integration of external devices into the company network
– Passive traffic analysis (information leaks such as VLAN tags, etc.)
– Testing of the isolation effect/inter-VLAN routing
– Trunking attacks
– Analysis of penetrability into other, including physically separated networks

In order to perform the test, SySS only requires one network connection to the company network. Ideally, the consultant receives a detailed list of configured network segments, including VLAN ID and IP addresses.

Tip by Sebastian Schreiber

You should ideally provide our consultant with a network plan showing every utilized network segment including the VLAN ID. This saves a great deal of time during a penetrability analysis.

2.4.6 PENTESTBOX: Security test via VPN

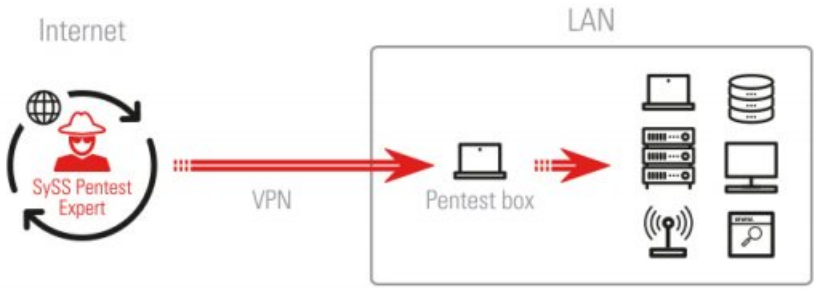

Figure 2.9: Module PENTEST BOX

SySS IT security consultants traditionally carry out the LAN/CLEAN and LAN/TRAINEE modules on site with the customer. In most cases, this is practical and warranted, primarily because it results in a physical meeting and cooperation, which usually leads to a better relationship with the customer and a positive project schedule on both sides.

However, it is not always practical and sometimes even impossible to be on site with the customer for the tests. This is why SySS has developed the "pentest box", which enables all on-site scenarios to be carried out externally via VPN. The pentest box, which the consultant uses to perform the test, is sent to the customer instead of an IT security consultant. This not only increases flexibility and reduces costs, but also contributes to a better environmental footprint. In general, every module that is usually carried out on site can also be performed using the pentest box. The pentest box can be used flexibly for targeted attacks against individual components using the TARGET/TECH module (see Subsection 2.7.1 on Page 48), as well as for testing individual client systems in the LAN/CLIENT module (see Subsection 2.4.3 on Page 35). The pentest box can even be used to test the security of an existing on-site wireless network (see WLAN module, Section 2.8 on Page 51).

In order to use the pentest box, the customer must first receive it. SySS sends a packet comprising a laptop, USB Ethernet adapter and power adapter to the customer. The customer receives a separate letter containing a smart card to unlock the encrypted laptop hard drive and instructions for starting up the pentest box. Full encryption of the hard drive ensures that no data is lost in the mail or can be picked up by third parties. The set-up is generally very easy for the customer and requires hardly any time – the laptop just needs to be connected to the mains using the power adapter supplied and started up. The notebook is unlocked once with the smart card and can then remain switched on.

The pentest box is connected to the customer's internal network, and the pentest box automatically connects to one of the servers controlled by SySS via another network (LAN, Wi-Fi or via mobile internet). Unlimited access to the internet or to the relevant SySS server is ideal for this other network to achieve the best possible connection. If it is not possible to establish an internet connection via LAN or Wi-Fi due to organizational or technical reasons, the pentest box will automatically attempt to connect to the SySS server via mobile internet.

Pre-configuration of the pentest box by a SySS consultant means that there is hardly any additional work for the customer to continue configuring the system or maintain it after initial start-up.

As soon as the VPN connection is established, the consultant can begin the test via the pentest box. The pentest box here behaves as would a laptop brought to an on-site test – it is therefore clearly visible to the customer.

As for other test modules, close collaboration between the customer and the consultant is always advantageous to address any technical and organization problems beforehand. In this way, the consultant is available for the customer in an advisory capacity at any time by phone or video conference if required, for example to initially start up the pentest box. Should there be any problems with the pentest box during the project, the customer can carry out "bug fixing" via a dedicated user under the guidance of a SySS consultant.

After the end of the test, all data is deleted by completely resetting the pentest box and the corresponding server. This resetting means that the pentest boxes cannot retrospectively establish a VPN connection. These measures guarantee data integrity – before, during and after the project.

In addition, the LAN/CLEAN and LAN/TRAINEE scenarios described above (see Subsection 2.4.1 on Page 34 and Subsection 2.4.2 on Page 34) and modules such as LAN/CLIENT (see Subsection 2.4.3 on Page 35) can be carried out. To do this, the customer sends a notebook to the SySS consultant who is performing it. This notebook is then returned by SySS via the customer's pentest box to their internal network. There is therefore the option to combine the LAN/CLEAN or LAN/TRAINEE module with the module LAN/CLIENT module since the pentest box must be with the customer for LAN/CLIENT.

2.5 VOIP-UC: Security analysis for Voice-over-IP and Unified Communication

Communication using Voice-over-IP (VoIP) has been common practice for years and can be found in many business sectors. Besides the traditional telephony, audio and video conferencing, chat functions, softphones, but also communication via the browser, for example, impose the current demands on this technology. This network of communication options is also known as Unified Communication (UC).

An attack on a VoIP/UC infrastructure can take place for various reasons. A successful interception of conversations can help attackers to take possession of highly critical and sensitive data. Industrial espionage should be mentioned in this context, which has by now become a common and extremely lucrative motive for attackers to detect vulnerabilities in VoIP in order to assess the competitiveness of their competitors, so that they can then either weaken it or strengthen their own.

In addition, VoIP systems are often integrated into the existing infrastructure and are not isolated from it. As a result, compromised systems can be used to spread further within the local network and take over other systems outside of the VoIP infrastructure.

SySS offers the test modules described below for the analysis of such systems.

Figure 2.10: Module VOIP-UC

2.5.1 VOIP-UC/INFRA: Security analysis for VoIP and UC infrastructures

Summary

During the security analysis, the internal VoIP and UC infrastructure is examined for security vulnerabilities and configuration weaknesses. Client and server systems are attacked with the aim of compromising them. Various machine-in-the-middle and eavesdropping attacks also attempt to spy on sensitive data, such as conversations. Potential attacks against neighboring systems and integrated services will also be examined.

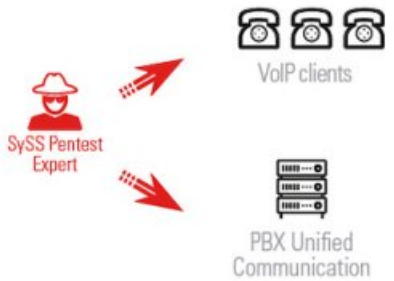

Figure 2.11: Module VOIP-UC/INFRA

Starting situation

Today's corporate communications no longer consist only of traditional telephones and a telephone system. Any application servers, automatic switching and in-depth integrations into existing systems are common practice, but at the same time increase the complexity immensely. This may affect the security level of the communication's infrastructure, which is why it is recommended to carry out a professional security analysis.

Objective

The VOIP-UC/INFRA test scenario is based on the risk posed by an attacker with access to a network connection in the corporate network. The aim is to uncover and assess any existing risks in the VoIP/UC client and server components and to make suggestions on how to resolve them. Not only are mere security gaps considered, but the configurations are also examined with regard to possible weaknesses.

Due to the large number of communication channels in such an infrastructure, not only the vulnerability of individual systems is assessed, but also communication between services to determine machine-in-the-middle vulnerabilities or other protocol-based attack potentials. In this respect, the ability to reconstruct conversations from recorded communication, for example, is of particular interest.

Implementation

As part of the audit, SySS primarily investigates the options for creating recordings of media and control connections and carrying out attacks against other identities as well as the possibilities of distribution into the adjacent infrastructure. In addition, vulnerabilities and configurations of involved systems are also thoroughly examined. In detail, the following points, among others, are taken into account in the VoIP analysis:

- Passive and active traffic analysis (signaling, configuration and voice data)
- Analysis of the central administration and provisioning system
- Network-based attacks against both VoIP clients and servers
- Attacks against a VoIP client with physical access
- Boot attacks against VoIP telephones, recording and evaluation of a boot process
- Eavesdropping attacks (machine-in-the-middle, logging functions in the integrated web server, etc.)

Cooperation by the customer

For the test, access to the test object must be defined (from the internet, from the intranet via a VPN, etc.). In addition, the network position from which SySS launches attacks must be defined.

Furthermore, SySS requires two VoIP clients configured by default (e.g., desk phones or softphones) and two configured Unified Communication clients from the company portfolio to carry out the test.

2.5.2 VOIP-UC/CLIENT: Security analysis for VoIP/UC clients

Summary

When analyzing the security of a VoIP/UC client, the software or hardware is checked for security issues. In addition to various machine-in-the-middle, injection and authentication attacks, this also involves attempts to exploit data available on the client, for example, to carry out attacks against participating server systems. Depending on the initial situation, various debuggers, disassemblers, fuzzers and application-specific tools such as HTTP proxies are used for implementation.

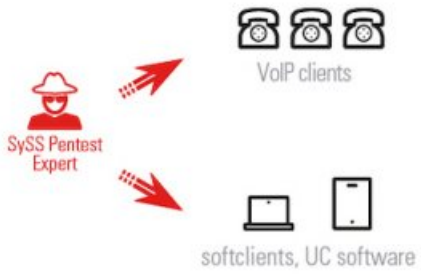

Figure 2.12: Module VOIP-UC/CLIENT

Starting situation

A VoIP/UC client is required to use VoIP and UC services and must be made available to users. The overall security of communication depends on the hardening of the client used. If device configurations are not protected or sensitive data can be extracted, this may pose a specific threat – and not just for the individual client. It only plays a subordinate role whether it is hardware – such as, for example, a desk phone – or software – such as UC software.

Objective

SySS takes the position of an attacker with access to a configured VoIP or UC client in order to investigate it. In addition to configuration hardening and implementation, this analysis also examines network communication and the services offered by the client for weaknesses. Central administration and device provisioning also take center stage. Options for privilege escalation or extending rights laterally are part of the analysis as well.

Implementation

The VoIP client (desk phone, softphone) provided by way of example is subject to a security analysis. Here, the following test steps, among others, are carried out:

– Traffic analysis
– Eavesdropping attacks
– Application-specific injection attacks
– Analysis of terminal services (HTTP, SIP, SSH, etc.)
– Provisioning analysis and centralized management

Cooperation by the customer

To carry out this test module, one respectively two preconfigured VoIP or UC clients to be examined including login data must be provided. Furthermore, network access to any server systems may be required during the project period to ensure full functionality of the clients.

2.5.3 VOIP-UC/CONF: Security analysis for audio and video conferencing systems

Summary

During the security analysis of audio and video conferencing systems, the components and technologies involved are checked for vulnerabilities and configuration weaknesses. For example, media encryption is examined and the question of the extent to which attackers can eavesdrop on sensitive content is clarified. It is also checked whether internal resources and services can be accessed through the mostly exposed systems and whether an effective authentication concept has been implemented.

Figure 2.13: Module VOIP-UC/CONF

Starting situation

Conference and UC systems have long been an established part of company-wide and comprehensive communication, be it classic video or audio conferencing or chat conversations. It is impossible to imagine modern communication without such systems. Since the information transmitted is often highly sensitive data, it is important to assess the security of such systems and to determine the security level in a detailed analysis.

Objective

During the security analysis of conference and UC systems, the implemented security measures are checked and an attempt is made to exploit the protocols and technologies used, such as WebRTC, STUN and TURN, for attacks against the systems. Furthermore, the implemented encryption algorithms and methods are examined to respond to the question of secure end-to-end encryption, for example.

Implementation

The conference system and the components involved are subject to a security analysis. This includes the following steps:

Encrypting media data: The encryption methods and their implementations for the transmission of media data (audio, video) are examined. Besides, the exchange of keys between conference participants and the media servers involved are analyzed and checked whether eavesdropping on conference content by third parties is possible (end-to-end encryption).

Authentication procedures: Among other things, it is checked whether unauthorized persons can gain access to conferences or whether existing conferences can be enumerated in an automated way.

Security analysis of server services: The server services are subject to a manual and (partly) automated security analysis. For example, potential web and VoIP services are checked for weaknesses. In addition, the STUN and TURN services required to establish connectivity between conference participants are analyzed.

Cooperation by the customer

To carry out the penetration test, SySS requires user accounts that are authorized to access the components to be tested. In addition, SySS recommends that available documentation about the implementation and involved software components be provided.

2.5.4 VOIP-UC/SBC: Security analysis for Session Border Controllers

Summary

The security analysis of the Session Border Controller (SBC) attempts to exploit potential security issues and configuration weaknesses, for example, to initiate unauthorized calls and eavesdrop on conversations. In addition, (application-specific) injection and authentication attacks are carried out against the SBC itself, which would result in a compromise of the system. For the best possible efficiency in execution, the current configuration is also considered in detail. Thus, implemented protective measures and, for example, the configured routing rules can be evaluated.

Figure 2.14: Module VOIP-UC/SBC

Starting situation

If heterogeneous VoIP services are connected and secured, the use of a Session Border Controller (SBC) is unavoidable. Its main function is to control and monitor signaling and media flows between different IP networks or domains. As a result, the SBC plays an important role in ensuring the security and integrity of communications services. Due to its central position and its function as a limiting point between different networks, the SBC is an attractive target for potential (external) attackers.

Objective

During the security analysis of the Session Border Controller, an attempt is made from an attacker's point of view to identify security measures and then bypass them in a targeted manner. For this purpose, the configured and mostly exposed interfaces of the SBC are determined and examined for vulnerabilities. Furthermore, the configuration for hardening and potential weaknesses are considered in the form of reviews.

Implementation

The protection mechanisms implemented in the SBC are checked for their effectiveness and attempts are made to circumvent them. The following protective functions may be investigated:

– SIP request normalization
– Security offloading
– Media anchoring/pinholing
– Topology hiding
– Allowlisting and blocklisting
– Error handling

For optimal results, it is also recommended to carry out a configuration review. The various couplings between the connected VoIP services and the authentication concepts are checked in particular. In addition, the configured encryption methods and algorithms are analyzed and evaluated.

Cooperation by the customer

To carry out the penetration test, SySS requires network access to the configured VoIP services of the SBC. It is recommended to provide detailed information about the connected and integrated VoIP services in order to gain maximum benefit during the test time. If the recommended configuration review is part of the project, read-only access to the system to be examined or a current configuration file shall also be supplied.

Tips by Sebastian Schreiber

1. Provide our consultant with a detailed overview of all VoIP and UC components involved and ideally an overview or architecture diagram of the conference infrastructure. Also make sure that required network and user accounts be applied for early on.
2. For the VOIP-UC/CLIENT module, please provide clients, configurations, and permissions that are as realistic as possible. It is also recommended to apply for the software/hardware, any necessary licenses, and the required accounts in good time so that no delay in project start is to be expected.
3. With the VOIP-UC/SBC module, please grant our consultant read-only access to the SBC or provide an up-to-date configuration export. In this way, potential vulnerabilities can be quickly identified and the test time optimally used.

2.6 SAP: Security analysis of SAP ERP environments

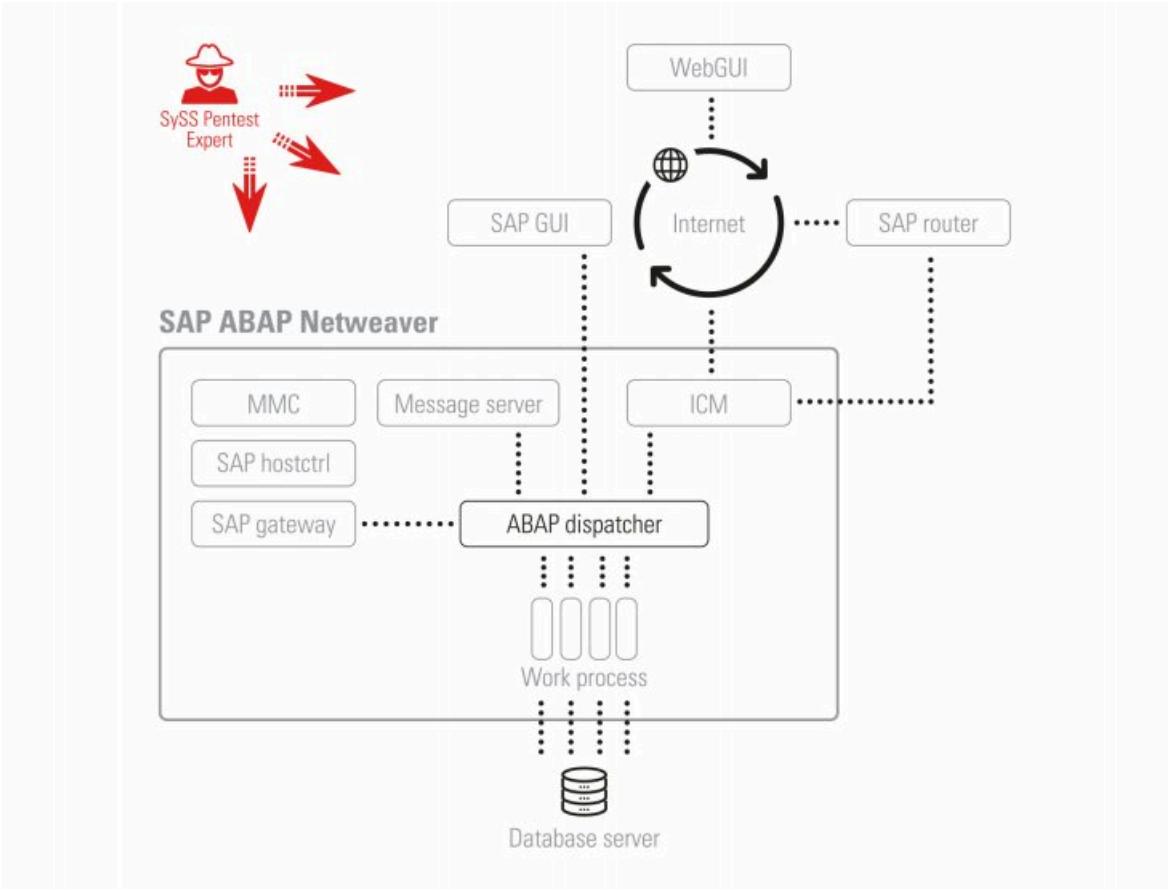

Figure 2.15: Module SAP

The SAP ERP product from the software manufacturer SAP is the market-leading solution for handling business processes such as bookkeeping, controlling, sales or human resources in companies of any size. SAP also provides opportunities to display these business processes individually and geared towards the individual needs of a company. The complexity resulting from specific requirements and individuality, as well as the criticality of the data processed, create a very lucrative target for a wide range of attackers, which attracted media attention in 2019 with the "10KBLAZE exploits".[2] Regular security tests should therefore be carried out, especially in critical environments such as SAP, in order to prevent economic espionage or sabotage attacks for instance.

As part of this security analysis, SySS provides different test scenarios or test objects which cope with the complexity of the utilized SAP software solution:

- Privilege analysis (based on roles, via SAP GUI/WebGUI)
- Analysis of the providing infrastructure
- Analysis of the utilized database connection and configuration
- Analysis of the SAP-specific configuration on client systems
- Analysis of the SAP system configuration (including, for example, SAP routers)
- Testing of the utilized SAP web application(s)

The procedure during the different test scenarios is normally organized very differently and ideally matches the individual customer requirements. The utilized tools also differ from the other modules, even if a test is performed via the internet. For example, it may be possible during such a test to access active services of the

[2] https://github.com/gelim/sap_ms/blob/master/10KBLAZE.md

Internet Communication Manager (ICM). A worst case scenario in this context would be total compromising of the internal SAP system, as well as privilege escalation and extension in the internal network.

During the internal security analysis, the focal points are primarily a system and configuration analysis, a privilege check of different SAP user roles and the providing infrastructure of the SAP environment. Where an SAP router is used, this also moves into the center of the system and configurations analysis. The above-mentioned privilege analysis of the provided SAP roles is normally carried out on the SAP GUI which is primarily installed on client systems. Testing of the activated WebGUI is only considered on rare occasions. A few roles used in the company are normally selected for this purpose. During this test, additional specific system parameters are identified and evaluated in the context of security technology, and possible risks are derived therefrom. SySS always needs a previously supplied administrative SAP user for this type of detailed test.

The special need to protect the providing infrastructure of a SAP environment and its different communication interfaces (RFC, DIAG or SOAP) must also be highlighted here as another important aspect. The services accessible during such a test, e.g. gateway, message server, management console or the ICM, are tested in regard to up-to-dateness and configurative errors. Another search is carried out for unauthorized ways to access highly critical company data, e.g. secret project information or personal data relating to employees, customers or service providers. Moreover, the opportunities for expansion to other SAP systems are also analyzed because close integration of different SAP systems enable an attacker to gain access to further systems from one compromised SAP system.
In addition to the actual SAP application server and its components, both the SAP-specific configuration of the Windows clients and the configuration of the utilized database solution (e.g. Oracle, MSSQL, DB2, MaxDB, SyBase and HANA) are analyzed thoroughly during the test.

In order to implement these test objects, SySS uses both different security and vulnerability scanners and exploit collections such as the Metasploit Framework. Software tools developed in-house by SySS are also used in a related context during security tests. Typical SAP tools such as SAP GUI, SQL clients, PySAP, Bizploit (or Sapyto) and other publicly accessible tools are also used.

Tip by Sebastian Schreiber

Regardless of whether the examination is holistic or randomized: Always make sure that the consultant performing the test is provided with an administrative SAP user. This is the only way to obtain a well-founded statement on the security level.

2.7 TARGET: Simulation of targeted attacks

The past few years have shown that companies have increasingly become victims of so-called "targeted attacks". The main objective of these attacks is no longer to randomly take over perimeter systems or penetrate the DMZ via vulnerabilities which exploit systems that can be accessed from the internet, but rather to compromise selected targets in the internal company network: i.e. clients. Attackers make use here of techniques such as social engineering, phishing, spear phishing, whaling or waterholing. Consequently, a simple e-mail with a dangerous attachment or calling up a supposedly harmless website often triggers compromising.

In order to measure a company's susceptibility to these realistic attacks, SySS provides two test modules. Firstly, the TARGET/TECH module evaluates what technical protective measures have already been implemented in order to make these attacks more difficult and how effective they really are in the event of an attack. Secondly, the TARGET/PHISH module can be used to determine whether these attacks have also already become anchored in the minds of the company's own employees. Both test scenarios are described in the following sections.

2.7.1 TARGET/TECH: Technical testing of protective measures

Summary

SySS tests the resistance of selected work environments of the customer (e.g. notebook, desktop PC or thin client/terminal server) to attacks from the internet. The protective measures implemented locally on the end device, but also the filters possibly installed on intermediate systems are evaluated in this respect.

Figure 2.16: Module TARGET/TECH

Starting situation

Installed on client systems are a large number of software components which can be used, for example, by browsers in the form of plug-ins and therefore started indirectly. It is precisely these components – especially if outdated versions are used – which represent a worthwhile target during targeted attacks. Prominent examples here include Adobe Flash, Word macros or the Oracle Java runtime environment. Taking over one or more client systems may lead to far-reaching compromising of the company network and also often represents the first step towards permanent infiltration in the case of an advanced persistent threat (APT).

Objective

This special case of an internal security test depicts a targeted attack on a selected client system. SySS tries to take over the available client system using well-known, adapted malicious code or its own malicious code. The first objective is to evaluate the performance of the already existing protective measures and then recommend additional hardening measures. As a side-effect of this test, it is shown how far existing gateway security solutions make it more difficult to carry out these attacks.

Implementation

SySS will adopt the perspective of both the attacker and the victim in this module. The test is carried out locally at the customer's premises for this purpose. Based on a reference client, the consultant will attempt, for example, to download the malicious code from a root server of SySS or send it via an e-mail attachment to the target system. The following are tested, for example:

- Utilized browsers and browser plug-ins
- Document viewer and media player software (e.g. prepared PDF files or Office documents with macros)
- Malware in e-mail attachments
- Third-party software such as Oracle Java
- Duping intermediate stations such as mail filters, AV gateways, URL filters, content inspection, etc.
- Circumvention of local protective measures such as UAC or utilized end point protection and anti-virus solutions (AV)

Cooperation by the customer

Preparation: The following preconditions must be fulfilled by the customer:

– Provision of a reference client (e.g. desktop PC, notebook or thin client)
– Provision of a standard user account with opportunities to access e-mails and the internet

Contact person: Just like the other modules, it is also important in this module to have a contact person available on demand in the short term during the test period.

Tips by Sebastian Schreiber

Targeted attacks should always be simulated by our consultants, who assume the role of the attacker as well as the victim. The consultant can work very efficiently using this procedure. This also provides the opportunity to deal with the scenario of an employee who triggers attacks as a result of incorrect behavior due to a lack of knowledge. It is therefore tested whether the technology also provides maximum protection in this case.

2.7.2 TARGET/PHISH: Simulation of a phishing attack

Summary

Within the remit of this module, SySS simulates a phishing attack on behalf of the customer and provides an anonymized, statistical evaluation of the returns as a result.

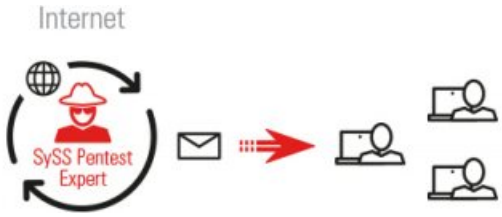

Figure 2.17: Module TARGET/PHISH

Starting situation

Phishing is not a new phenomenon. Every user is confronted with this type of attack nearly every day. Whereas normal "invoice e-mails" are rather wide-ranging and should reach "the masses" rather than specific addressees, customized e-mails are sent to selected recipients during targeted phishing attacks on a certain company. These e-mails are composed for certain employees in a deceptively genuine manner in order to increase the likelihood of a successful attack. This action is called spear phishing or whaling if the addressees mainly comprise "big fish" in the company.

The contents of these targeted phishing mails are often attachments infected with malicious code – for example alleged invoices – or links to websites on which the recipient is instructed to enter his/her access data. The objective of the attackers is normally to gain control of individual user accounts or even systems within the company in order to use them to copy, for example, sensitive intellectual property of the company and thereby gain a competitive advantage. Industrial espionage and spying actions are also not a new phenomenon. Press reports on phishing attacks are published continuously in the media and make this problem a matter of public interest.

Objective

Within the scope of this module, SySS will carry out such a phishing attack against the commissioning company. It is especially important here to emphasize that the aim is not to uncover misconduct by some individuals. On the contrary, SySS will carry out the attack in the most anonymous way possible and provide results in the form of a quantitative, statistical evaluation of the outcome of the attack (see Figure 2.18). The customer can use the results, for example, to increase sensitization in security awareness events.

Implementation

The procedure for this module is based roughly on the following pattern:

- The customer provides SySS with a certain number of e-mail addresses (e.g. 200 recipients).
- SySS prepares the phishing attack. For this purpose, information about the company is compiled, the contents of the e-mail are designed and any phishing pages are created.
- SySS sends the e-mail to a random selection of the supplied e-mail addresses.
- After a certain period of time (normally a few days), SySS will evaluate the results and send them to the customer in the form of anonymous, statistical documentation.
- Phishing is also possible via text message. SySS can use any telephone number as a sender in this case.

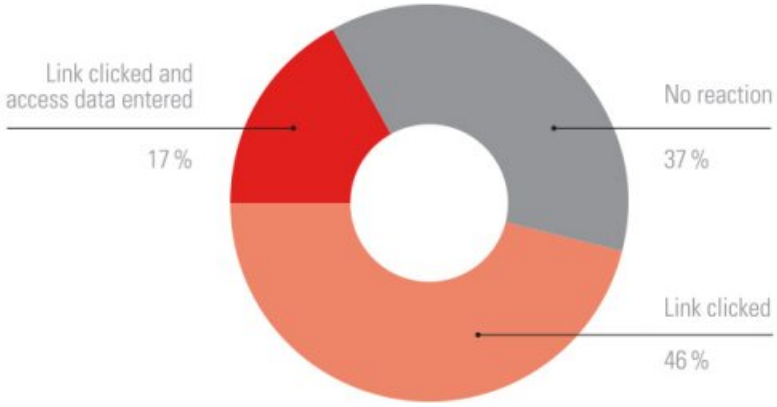

Figure 2.18: Example of the results of a phishing attack

Cooperation by the customer

Preparation: The preparatory tasks of the customer for this module are very clear. SySS must only be sent the list of the e-mail recipients (see "Implementation"). The customer of SySS may also like to provide information which increases the credibility of the phishing e-mail.

Contact person: SySS recommends that the phishing attack be played through beforehand with the customer's contact person. For example, this can prevent the attack from failing due to the possible use of filter mechanisms and the e-mails from not actually reaching their recipients. These eventualities should definitely be discussed jointly before the module is implemented.

The customer should also make preparations for the phishing attack being noticed in the company and for questions being asked. It is all the more important firstly to severely limit the length of the attack, and secondly to provide an explanation and an all-clear signal immediately after the attack!

Just like in the other modules, it is obligatory for the contact person to be available during implementation of the module. For example, unforeseen problems could arise when the phishing e-mails are sent. The contact person should also always be available to the recipients if they have any questions.

2.8 WLAN: Testing the wireless network

Summary

The Wi-Fi infrastructure of the customer is tested locally for security vulnerabilities. Client security can also be tested, for example in regard to a connection with so-called "rogue access points". Isolation of different Wi-Fi areas from other internal network areas can also be tested.

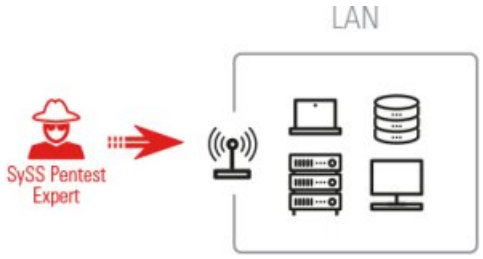

Figure 2.19: Module WLAN

Starting situation

Unlike wired networks, Wi-Fi can be accessed and received at any time by third parties, often also from outside the company's own business premises. This results in the risk of misuse of the Wi-Fi infrastructure. The risk is due to the fact that Wi-Fi is used without authorization and also that the transmitted data can be intercepted by unauthorized persons. This then affects the company network area which can be accessed via Wi-Fi.

Objective

In order to exclude the above-mentioned risks, both the access points and the Wi-Fi clients (e.g. notebooks or cellphones) are tested. The main subjects of the test are the utilized encryption and authentication methods, as well as the client configuration. In the case of the client configuration, one of the focal points of the test is resistance to machine-in-the-middle attacks. It can also be tested whether clients would also be connected with so-called "rogue access points".

A Wi-Fi visibility analysis can also be carried out, for example, by walking around the company's premises. In addition to visibility, Wi-Fi parameterization is determined during this analysis.

Implementation

The Wi-Fi test is carried out locally. The exact procedure depends to a large extent on the location to be tested and the utilized Wi-Fi infrastructure. The test procedure is based roughly on the following pattern:

- Inventory and parameterization: What is visible, what belongs to the customer's infrastructure?
- Verification: Do the findings correspond to the expectations and information?
- Detection of access points
- Testing of the networks for authentication and encryption
- Attack against fixed authentication and encryption
- Testing the Wi-Fi clients

Denial-of-service attacks are an inevitable component of the analysis of wireless networks. However, they can be restricted to selected systems (e.g. a reference client). A selection of possible test tools is presented under "Tools" (see Subsection 1.1.6 on Page 12).

SySS tests Wi-Fis here at 2.4 and 5 GHz based on IEEE Wireless Standard 802.11. Other wireless networks (based on DECT, UWB, IEEE 802.15.4, Z-Wave, Bluetooth, etc.) do not form part of a Wi-Fi test.

Cooperation by the customer

Preparation: Information about the Wi-Fi infrastructure, especially the utilized access point type and the nature of authentication, should be provided before the start of the test, preferably during the KICKOFF. In addition, all participants and Wi-Fi systems managers should be informed about the test and its intention, and should be available to answer questions during the test.

Contact person: Since Wi-Fi tests are always carried out locally, a contact person must always be available just like in an internal test. The contact person should be easy to reach during the test period and also enable the consultant to access the buildings to be tested. Past experience shows that the most efficient method is when the contact person personally holds these access privileges and can also issue them.

If the consultant is to test locations unaccompanied, one person for each location to be visited should be present in order to provide access to buildings. This person should also have been informed about the test and/or the visit. Optionally, the consultant can be supplied with relevant documents.

Access to buildings and the site: The permits required for this purpose must be available at the start of the test. This applies both to access for the consultant himself/herself and his/her equipment.

When the test period is chosen, special attention should be paid to opening hours and general working hours. If on-site inspections are necessary, they should only be carried out in daylight. If Wi-Fi clients are tested, reference clients should be available or samples should be selected. If several locations are to be tested on one day, factors such as traffic flow, etc. must be taken into account when choosing the test period and the duration of the test.

Tips by Sebastian Schreiber

Obtain the necessary permits for access to buildings in good time and inform the participants. This will prevent long and unproductive waiting times. Informing the participants helps them to regard the test as a positive measure and not as an disruptive inspection. If your workforce contains some people with misgivings, simply invite them to attend the test.

2.9 MOBILE: Security testing of mobile end devices, apps and mobile device management solutions

Mobile end devices such as smartphones or tablets have also increasingly made inroads into corporate IT in the last few years. In addition to e-mails and contacts, other critical company data is also synchronized at times on the devices, which therefore become another interesting target for attack. SySS therefore also offers various test modules in this area in order to evaluate whether the data on the customers' devices is well-protected against any attacks. Typical test scenarios include, for example, an analysis of the device configuration (MOBILE/DEVICE), a security test of utilized mobile apps (MOBILE/APP) or an evaluation of the mobile device management solution used for administration (MOBILE/MDM).

2.9.1 MOBILE/DEVICE: Security testing of mobile end devices

Summary

Selected mobile end devices – especially smartphones or tablets – are tested from different perspectives for security vulnerabilities which enable an attacker to access locally stored data without authorization and gain access to the company network via the mobile end device.

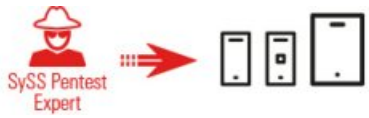

Figure 2.20: Module MOBILE/DEVICE

Starting situation

Sensitive information such as personal data or business-critical documents (e-mails, PDF/Office files, etc.) are often stored on mobile end devices including smartphones or tablets. These mobile devices also frequently provide access to the company network in order to be able to use, for example, PIM (personal information management) functions. These functions include, for example, synchronization of e-mails, contacts and appointments. This fact makes mobile end devices an interesting and worthwhile target for attackers. Since the devices are used constantly by people on the move, there is an increased risk of them being stolen.

Objective

During the security test, the selected mobile end device is tested for vulnerabilities from different perspectives. Depending on the focal point of the particular security test, attack scenarios are implemented from the perspective of an external attacker and/or from the perspective of an authorized user of the device. The objective is to show hardening measures.

Implementation

The mobile end devices are tested for security vulnerabilities in the SySS laboratory. Different tools and methods are used depending on the device type, operating system and focal point of the test. Possible attack scenarios are described below.

Attacks from the perspective of an external attacker:

- Attacks via network interfaces of the device (Wi-Fi, Bluetooth)
- Attacks against network services
- Machine-in-the-middle attacks against utilized apps (e-mail synchronization, VPN access, document management, etc.)
- Attacks with physical access to the device (theft scenario): non-authorized access to locally stored data; manipulation of the device (for example, installation of malicious software)

Attacks from the perspective of an authorized user:

– Attacks on data of external users via PIM functions
– Manipulation of the device (e.g. jailbreak or rooting, installation of non-approved apps)
– Security analysis of selected apps (document management, remote access to systems in the company network, mobile banking)

Implemented protective mechanisms, e.g. PIN/password/biometric login, deletion of user data after a certain number of failed attempts to log into the device, detection of so-called jailbreak or rooting attempts, or remote resetting of devices (remote wipe), which are provided, for example, by utilized mobile device management solutions can also be verified during the security test.

Cooperation by the customer

Test preparation: Before the start of the test, information about the hardware to be tested, the operating system version and relevant attack scenarios should be given during the KICKOFF. All participants, e.g. system managers for e-mail servers or mobile device management servers, should also be informed about the security test and be available to answer questions during the test period.

Contact person: In order to carry out the test on a mobile end device, logistical preconditions must be fulfilled in particular. For example, transportation of the devices must be organized and, just like in other tests, a contact person must also be available to answer questions.

2.9.2 MOBILE/APP: Security testing of mobile apps

Summary

The applications (mobile apps) that can be installed on a mobile end device are tested for vulnerabilities during this security test. Mobile apps are analyzed statically and dynamically using decompilers and debuggers and the locally stored data is also analyzed. Runtime manipulation tools are used, too. The data traffic to the server is analyzed and encryption is thoroughly tested as part of a machine-in-the-middle attack. In addition to testing of the back-end servers accessible via the internet, and of the connection of the mobile app to these servers, security settings within the mobile apps are analyzed and evaluated in accordance with the required protection needs.

Figure 2.21: Module MOBILE/APP

Starting situation

Sensitive data is often stored on mobile end devices, such as smartphones or tablets. Since a large number of security measures are often self-implemented during in-house development of mobile apps, SySS recommends analyzing the apps before they go live – especially when personal data or information requiring a high degree of protection are processed by the corresponding mobile apps. Another reason for an extensive security analysis of mobile apps – even if they were not self-developed – is the possibility of an attack against internal company resources (e.g. VPN functionality or similar) using mobile apps.

Objective

During the security test, the mobile apps are tested as agreed for vulnerabilities in the respective version developed for the corresponding application store (e.g. Android or iOS). In addition to compliance with the security requirements, data traffic is monitored. On request, mobile apps are also tested from different perspectives. Finally, SySS estimates the security level and proposes measures to eliminate or reduce any vulnerabilities.

Implementation

The mobile apps are tested for security vulnerabilities in the SySS test laboratory. Different tools and methods are used depending on the focal point of the test. The mobile apps to be tested are installed, for example, on a device with jailbreak or root privileges in order to have full access to the file system and stored content during the analysis. If necessary, it is evaluated beforehand whether any utilized rooting/jailbreak detection can be circumvented. The mobile app is then decompiled, for example, and is subjected to a static code and dynamic runtime analysis. Other possible attack scenarios include machine-in-the-middle attacks or other traffic-based attacks on data transmission between the mobile app, other apps and its server back end. Attention can also be paid to potential infringements of privacy. Depending on requirements, individual tweaks are developed to circumvent security measures.

Cooperation by the customer

Depending on the test scenario, the following requirements must be fulfilled in order to test mobile apps:

Test preparation: The mobile app to be tested must be provided by the customer in the version to be tested (for each operating system to be considered) if it is not available via an application store. Furthermore, some mobile apps cannot be used until the mobile app has been successfully authenticated. SySS requires corresponding user accounts for tests from this perspective. Before the start of the test, it is also practical to inform the consultant, e.g. during the KICKOFF, about organizational and technical dependencies, as well as the mobile app to be tested (e.g. documentation) and relevant attack scenarios.

Contact person: Since the mobile app is ideally analyzed in the SySS test laboratory, the person responsible for the app to be tested should at least be available by phone during the test period. In order to answer detailed questions, it is useful to have direct contact with the developers or technical contact persons.

Tip by Sebastian Schreiber

If you want to offer a mobile app for several operating systems – for example, iOS and Android – and have it tested, you must inform the consultant in advance about possible interfaces so that no valuable test time is lost. A classic example of this is a web service interface used jointly by different mobile app versions.

2.9.3 MOBILE/MDM: Security testing of mobile device management solutions

Summary

This security test is used to evaluate the mobile device management solution (MDM) used to administer mobile end devices such as smartphones or tablets. During the security test, SySS firstly analyzes the server infrastructure – this also includes the primarily web-based management interface and other network services provided by the MDM solution. Secondly, the security settings rolled out by the MDM solution are evaluated according to the required level of protection.

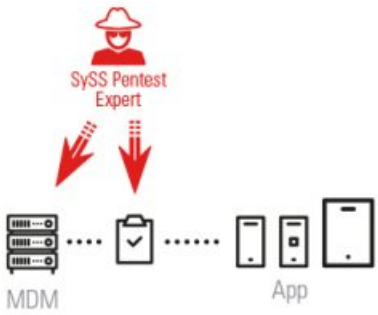

Figure 2.22: Module MOBILE/MDM

Starting situation

A mobile device management solution (MDM) is normally used to integrate mobile end devices into the company network and manage the data stored on the mobile devices according to security guidelines. To ensure that the MDM-administered devices can receive updated configurations, for example, from any location, the MDM servers are also accessible from the internet. Since a large volume of business-critical data can also be accessed via these servers, MDM servers represent a valuable attack target. The MDM-administered device configurations are business-critical since they have a direct effect on the level of security of the mobile end devices.

Objective

Within the framework of this test module, SySS initially tests the infrastructure which connects the mobile end devices to the company network. For example, both access to the e-mail server and the remaining data traffic are analyzed in regard to compliance with the security requirements. The registration and provisioning process, as well as the associated check on whether the mobile device complies with company guidelines are analyzed. SySS primarily examines the question of whether a device whose security was compromised can be connected to the company network. For example, jailbreak/rooting detection implemented in the MDM solution is analyzed

for this purpose. Another objective is to carry out a critical evaluation of the rolled out device configurations. In this case, it is necessary to make recommendations which reduce the risk posed by a compromised device.

Implementation

The penetration test of the MDM solution can either be carried out via the internet at the SySS laboratory or directly at the customer's premises. SySS tests the complete "life cycle" from provisioning through to decommissioning of the device. An attempt is made here to show vulnerabilities in the implementation of the security guidelines. Special security needs and requirements of the customer, as well as special test scenarios can also be examined in this case. If, for example, there is a requirement that mobile devices may only be operated in a clearly defined state, SySS tries to compromise this state.

Within the framework of data loss scenarios, there is an evaluation of interfaces through which data can leave the company. In this case, SySS examines the question of which precautions can be implemented to ensure that an employee cannot negligently copy company data to another insecure device. These vulnerabilities are shown if a container solution provides corresponding export functions or backup can be carried out on a private computer.

To sum up, the following test scenarios are conceivable for instance:

– Network-based security analysis of the participating MDM infrastructure servers
– Web-based security analysis of the management interface
– Security evaluation of the configuration profiles and guidelines
– Vulnerability analysis of the MDM app
– Analysis of processes: provisioning, configuration update, remote deletion, deprovisioning, etc.
– Traffic analysis, identification of protocol-based vulnerabilities

Cooperation by the customer

Test preparation: Depending on the test scenario, the following requirements must be fulfilled in order to test the MDM infrastructure:

– Approval to test the MDM servers via the internet must have been granted.
– It must be ensured that the MDM profiles/guidelines can be examined.
– An opportunity must be available to personally set up test devices.
– The MDM app to be tested must be provided, unless it is available in the respective app store.

Contact person: The test of the MDM infrastructure has numerous overlaps with other areas. The contact person responsible for protecting the MDM server, which is accessible via the internet, should be included along with the contact person for the local network and Wi-Fi. As soon as private devices (keyword: "bring your own device", BYOD) are also involved, the processes should be agreed with the Works Council. Since short-term availability problems cannot be excluded during the test, SySS recommends that all users of mobile devices be informed – this also helps to create security awareness.

2.10 CLOUD: Security analysis and hardening measures for cloud services

Cloud computing has been an important trend in recent years. The fact that cloud services combine server capacities and apparently unlimited storage space makes them very attractive to users. In addition, easy access to other clients via the internet, as well as users' own computers, smartphones and tablets, contributes to the growing popularity of cloud computing. However, there is a snag to this practicality – whenever there is a

new development, hackers are generally not far away from developing attacks against cloud services. With the CLOUD module, SySS offers security tests of the Amazon Web Services environment and Microsoft Azure infrastructures. The features of this test module are described in detail in the following sections.

Figure 2.23: Module CLOUD

2.10.1 CLOUD/AWS: Security analysis for Amazon Web Services projects

Summary

The cloud infrastructure and services utilized within it are tested for vulnerabilities and possible hardening measures during a security analysis of an AWS environment, which comprises a configuration audit and elements of the penetration test.

Figure 2.24: Module CLOUD/AWS

Starting situation

Hardly any companies can do without cloud services any more when it comes to optimizing costs and scaling options. One of the major providers in this field is Amazon, with Amazon Web Services (AWS). Irrespective of whether a cloud project has just been constructed or is already operating, a security analysis detects vulnerabilities and project-specific hardening measures improve the security level of the cloud project. In addition to fundamental functions, such as testing the role and privilege concept, as well as the key management for protecting sensitive data, SySS also checks storage facility privileges, such as for S3 (Simple Storage Service) or databases. If the infrastructure is dynamically generated using "Terraform" or "CloudFormation", SySS checks the security of the systems to be used in the future. In addition to the templates, the security of the image sources and Auto Scaling Groups is tested.

An extensive network configuration, comprising VPCs, subnetworks, security groups and gateways are also analyzed for open attack surfaces and potential improvements are formulated.

The auditing of organizational and technical topics, such as monitoring, logging and alert workflows, also falls within the scope of SySS. For example, checks are undertaken to determine whether the correct employees receive the correct security messages and whether these are cluttered with unnecessary details.

If the project is to be built on a serverless infrastructure, SySS evaluates the customized implementation using services such as AWS Lambda, AWS DynamoDB or AWS Cognito.

For an IoT project, the enrollment process – the communication with the IoT Core and the configuration of the corresponding rules (topic rules) – is analyzed such that the devices and their users are configured according to the principle of least privilege.

Objective

Together with the customer, SySS attempts to lift the cloud project to a high security level or to confirm such a level. In addition to a catalog of vulnerabilities, SySS formulates a list for recommended and possible hardening measures.

Implementation

Cloud audits are typically performed remotely. SySS is provided with a user with read permissions for the relevant project. This approach is supplemented by phone interviews with competent persons in order to clarify any unanswered questions or to offer an insight into the organizational structure. It should also be determined, for example, whether there are topics which have not yet been illustrated in the project. Examples of this are vulnerability scans of instances or a missing emergency account in the event of multi-factor equipment failure.

Cooperation by the customer

An audit can only assess those elements which can be viewed. It is thus extremely important that the necessary read permissions are set for the project.

It is therefore worth checking, prior to handover of the auditor accounts to SySS, whether the established permissions will suffice to see all of the relevant factors.

The combination of the following privileges has proven to be particularly effective:

- `arn:aws:iam::aws:policy/job-function/ViewOnlyAccess`
- `arn:aws:iam::aws:policy/SecurityAudit`

In addition to the auditor privileges, which allow users to have a "top-down" view, it is also worthwhile to provide user accounts for each project so that scenarios can be tested from a user's perspective.

2.10.2 CLOUD/AZURE: Security analysis for Azure infrastructures

Summary

The cloud infrastructure and services utilized within it are tested for vulnerabilities and possible hardening measures during a security analysis of the Office 365 environment, which comprises a configuration audit and elements of the penetration test. SySS also evaluates the security level of the Office 365 configuration.

Figure 2.25: Module CLOUD/AZURE

Starting situation

Nowadays, it is ubiquitous for companies to have their IT structure based on Microsoft Active Directory, Windows servers and Windows Office clients. In order to more dynamically scale this infrastructure and to make remote working easier, entering the Azure cloud is an option for meeting requirements for greater flexibility. Irrespective of whether customers are just making their first steps in the Azure cloud, have already implemented major relocations and projects or even choose "cloud only", SySS will assist them in raising their Azure infrastructure to a high security level or, in a best-case scenario, confirm that the Azure environment is already very well protected.

The Azure Active Directory is generally analyzed during the first stage of the Azure testing. Checks are undertaken to ensure that the user and group privileges, as well as authentication, are such that they fulfill the customers' security requirements. Whether customers build and expand their infrastructure using virtual machines, Kubernetes or without a server entirely, SySS ensures that no configuration errors result in vulnerabilities or even data loss. In an audit which encompasses manual and automated tests, each configuration is tested for security-related incorrect configurations and use of best practice methods. SySS also carefully examines the monitoring, logging and alert workflow and provides an overview of what is possible and what is necessary to strengthen the Azure environment in the long term.

An Office 365 configuration audit primarily concerns identifying whether employees' data and e-mails are protected against unauthorized attacks among themselves and by third parties. The following aspects, for example, are examined in this case:

- User privilege settings and distribution of roles in Office 365 and Azure AD audit of Office Secure Score use and OneDrive configuration
- Data storage and attachment testing, e.g. storage of critical data and malware detection
- Data flow monitoring testing (OneDrive, SharePoint, etc.)
- Testing of login monitoring or attack monitoring/notification of administrative actions (e.g. the correct use of Azure Advanced Threat Protection)

SySS also shares its expertise when carrying out a security analysis of Azure IoT environments. The following topics are discussed:

- Enrollment, registration and authentication process for the IoT device
- Testing of data communication between the IoT device and Azure IoT hub for vulnerabilities
- The safe use of Azure Event Hubs and Azure Functions
- Suitable monitoring for detecting compromised devices
- The best practice use of the Azure IoT SDK on devices
- Testing a potential client separation of the IoT landscape for the possibility of privilege escalations

Objective

Together with the customer, SySS attempts to lift the cloud project to a high security level or to confirm such a level. In addition to a catalog of vulnerabilities, SySS formulates a list for recommended and possible hardening measures.

Implementation

Cloud audits are typically performed remotely. SySS is provided with a user with read permissions for the tenant or project resources and the AAD. This approach is supplemented by phone interviews with competent persons in order to clarify any unanswered questions or to offer an insight into the organizational structure. It should

also be determined, for example, whether there are topics which have not yet been illustrated in the project. Examples of this are vulnerability scans of instances or a missing emergency account in the event of multi-factor equipment failure.

Cooperation by the customer

An audit can only assess those elements which can be viewed. It is thus extremely important that the necessary read permissions are set for the project.

It is therefore worth checking, prior to handover of the auditor accounts to SySS, whether the established permissions will suffice to see all of the important configurations.

A user with the "Global reader" privileges should be provided to the tenant, depending on the project scope. In addition, it should be possible to grant privileges to the project-related resource groups.

In addition to the auditor privileges, which allow users to have a "top-down" view, it is also worthwhile to provide user accounts for the project so that scenarios can be tested from a user's view.

2.10.3 CLOUD/GCP: Security analysis for Google Cloud Platform environments and Google Workspace

Summary

The security analysis of a Google Cloud Platform (GCP) environment, which includes both a configuration audit and elements of penetration testing, checks the cloud infrastructure and its services in use for weaknesses and possible hardening measures. The security level of a Google Workspace configuration will also be evaluated.

Figure 2.26: Module CLOUD/GCP

Starting situation

A further provider in this area is Google with its Google Cloud Platform (GCP) and Google Workspace. Regardless of whether a cloud project is currently being set up or is already in operation: A security analysis reveals weak points and project-specific hardening recommendations improve the security level of the cloud project. In addition to basic functionalities such as reviewing the role and authorization concept as well as the key management to protect sensitive data, SySS also checks storage permissions, for example from cloud storage or cloud SQL databases. If the infrastructure is created dynamically using "Terraform" or "Cloud Build", SySS assesses the security of the systems that will be used in the future. An extensive network configuration – consisting of VPCs, subnets, security groups and gateways – will also be analyzed for open attack surfaces and potential for improvement will be developed.

Moreover, the auditing of organizational-technical topics such as monitoring, logging and alert workflows falls within the scope of SySS. For example, it is checked whether the right security alert messages are received by the right employees and whether the latter are not unnecessarily overwhelmed with too much information.

If the project is based on a serverless infrastructure, SySS examines the tailored implementation with services such as Cloud Run, Cloud Functions and App Engine.

In an IoT project, the enrollment process, communication with the IoT Core and the configuration of the corresponding rules (Topic Rules) are analyzed so that the devices and their users are configured according to the principle of least privilege.

Objective

In cooperation with the customer, SySS aims at either enhancing the cloud project to a high level of security or confirming this very one. In addition to a catalog of weaknesses, SySS also develops a list of recommended and possible hardening measures in a differentiated way.

Implementation

Cloud audits are usually carried out remotely whereby SySS obtains a user with read permissions to the Google Cloud project and cloud resources. This approach is supported by telephone interviews with responsible persons in order to clarify open questions or to gain insight into the organizational structure. It should, for instance, also be determined whether there are topics that have not yet been covered in the project. Examples include regular vulnerability scans of instances or a missing emergency account should there be a failure of the multi-factor devices.

Cooperation by the customer

In an audit, only what can be viewed can be evaluated. It is therefore particularly important that the required read permissions are set for the project. Before handing over the auditor accounts to SySS, it is beneficial to check whether the authorizations assigned are sufficient to be able to view all important configurations. Depending on the scope of the project, a user should be provided with the permissions of a "viewer" within the Google project and the Security Reviewer, but also with read access to the Security Center. Furthermore, read permissions should be provided for project-relevant resource groups.

In addition to auditor authorizations, which enable a "top-down" view, it is also useful to provide user accounts of the project so that scenarios can be checked from the perspective of an unprivileged user.

Tip by Sebastian Schreiber

The cloud is more than relocating virtual machines. It is a ecosystem with its own rules and which particularly relates to security. Major projects can be realized in a very short time and are often implemented by a company team without the traditional background in IT. You should therefore also take care to ensure that security plays an important role in this ecosystem.

2.11 EMBEDDED: Security analysis of embedded systems

Summary

With its Embedded Security (ES) modules, SySS offers wide-ranging security analyses of the various embedded systems. These extend from analysis of the externally accessible interfaces via cable or wireless transmission to investigation of the internally installed components and the software used there. Proprietary sub-modules can also be offered for the analysis of individual products or control units from the automotive sector.

General prerequisites and recommendations for the implementation of the test modules are outlined below. These are then described in more detail, and possible attack scenarios are presented.

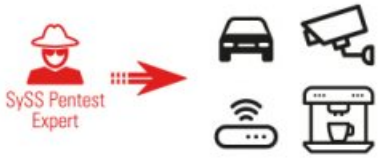

Figure 2.27: Module EMBEDDED

Starting situation

Sensitive information, such as credentials for a back-end system, client certificates or passwords for maintenance access, is often stored on hardware components. Internal functionalities – e.g. algorithms which process sensor data – are also often regarded as company secrets. Since the devices have not been normally subjected to a physical inspection by the vendor, third parties can use attacks to try and extract stored data or manipulate the device in another way.

Objective

By agreement, the objective of the security test is to test the hardware, its available interfaces, connections to the back end and the firmware itself for vulnerabilities. On request, the back end can also be analyzed. The back end is often implemented in the form of a web service interface, e.g. by means of SOAP or REST (see WEBSERVICE Section 2.3 on Page 29). Depending on the type of hardware, the objective may be, for example, to verify whether the data stored or transmitted there is adequately protected against external attacks. Protection of firmware against unauthorized copying or manipulation may provide motivation for such an analysis.

Implementation

If possible, the hardware components are tested in the SySS laboratory to check for security weaknesses. Different tools and methods are used depending on the test module. For example, individual chips such as memory elements may be selected and analyzed during invasive tests. The weaknesses shown below represent examples of vulnerabilities which are often discovered during these tests:

– Firmware extraction: The hardware does not have any effective measures which provide protection against firmware extraction.
– Maintenance access: Data can be extracted or the operation of the device can be manipulated through maintenance services, e.g. console access via a serial interface.

– Trivial passwords: Credentials for the bootloader or maintenance interfaces are well-known or can be easily guessed.
– Unencrypted memory elements: Unencrypted memory elements can be detected and then read out without restriction.
– Login data is stored in plaintext: The credentials stored in the hardware to be tested are particularly worthy of protection since they can be used for other attacks. However, extraction may be possible if protective measures (e.g. for the operating system or firmware) can be circumvented.
– Machine-in-the-middle attack on a connection: If encrypted connections are not properly implemented, an attacker can intercept and modify data traffic in a suitable position in the network.
– Replay attacks: Known actions can be triggered if already recorded communication is resent.
– Static key: Extraction of the static key material can result in the encryption being broken.
– Violation of the principle of least privilege

The specific implementation of the individual specific modules is explained from Subsection 2.11.1 on the following page.

Cooperation by the customer

During the penetration test, contacts should be available by telephone or by e-mail. This includes contacts for setting up the system, interface developers and supervisors of any further services with which the device communicates. Since the penetration test can also open up questions about the hardware itself, contacts should also be appointed in this regard.

Test preparation: SySS recommends providing at least two versions of the device to be tested during the penetration test. This allows for testing of whether an attack can also be transferred to other devices or whether other devices can be influenced by an attack vector. It also covers potential failure of the device. If the ES/INTERNAL module (see Subsection 2.11.4 on Page 67) is planned, a third version should ideally also be available. The devices should be provided to SySS several days before the start of the test. This enables the expected functionality of the test devices to be checked in advance.

To improve the test quality, SySS recommends providing all documentation on commissioning and the available interfaces and processes. Architecture diagrams, (protocol) specifications or technical manuals and hardening guidelines are also helpful. In addition, to save time, any test tools/software used by the customer, or previous lessons learned, can be shared with SySS.

Before beginning the penetration test, initial configuration of the device should be carried out in a joint workshop. Equally, an introduction to the use and configuration of the devices is also advisable. The workshop can be held jointly in the SySS lab, at the customer's premises or via a telephone or video conference. It should take into account not only the end user side, but also the provisioning and administration of the devices.

When using external services (Cloud), it can be necessary to obtain approval for the penetration test from the corresponding service provider. These questions can be clarified during the initial kick-off meeting (see Subsection 1.3.1 on Page 17).

Contact person: Since the hardware components are ideally tested in the SySS laboratory, the person responsible for the hardware component to be tested should at least be accessible by phone during the test period. In order to answer detailed questions, it is helpful to have a direct connection to the developers or another technical contact person.

Dependencies: SySS should be notified of any organizational and technical dependencies. This can be done during the kick-off discussion. If, for example, the test version is not a fully autonomous system, which can be tested separately from other systems, then SySS must be informed which other systems the test object is dependent on. SySS must also be informed of any test restrictions (time, technical, organization).

Tips by Sebastian Schreiber

Estimate the time required for a hardware test very generously! Check the systems with which your product communicates. A full test of these systems is normally beneficial if they already form an organizational and technical unit. Then select the suitable test modules for the test.

2.11.1 ES/AUTOMOTIVE: Security analysis of control units and sensors

Control units and sensors in the automotive area process data and control the vehicle's systems based on this data. Secure and fault-free operation of the systems can be vital. The issue of Intellectual Property is also very important here. For example, performance curves and engine control algorithms are particularly valuable to automotive manufacturers. Battery cell charge management can also be of significant importance in the field of e-mobility.

Figure 2.28: Module ES/AUTOMOTIVE

With the ES/AUTOMOTIVE sub-module, SySS offers an IT security test of the above-mentioned components. Test objects can be test setups with several components or even entire vehicles. Individual components such as ECUs, head units, sensors, etc. can also be subject to a security check. The procedure for this module is very similar to the procedure for the ES/EXTERNAL (see Section 2.11.2) and ES/INTERNAL (see Section 2.11.4) modules, but special protocols and technologies are often used in the automotive area, which have been developed specifically for this area. SySS therefore adapts the tools and attack techniques used accordingly. There are also differences in the classification of the security gaps found. These are described and evaluated with a special focus on the automotive industry.

Questions

During the penetration test, SySS answers the following questions, amongst others:

- Analysis of the diagnosis function: Are the diagnosis protocols used adequately protected? (Keyword: Security Access)
- Weakness in the bus communication (CAN, FlexRay, LIN, Ethernet): Which protocols are used and how are they configured?
- Tamper Detection of the ECU: Can the ECU be opened and manipulated without being noticed by the system?
- Can sensitive data – such as engine control data – be read out from the ECU?
- How do the actuators react to incorrect sensor data?
- Can the firmware be extracted and modified?

2.11.2 ES/EXTERNAL: Security analysis of cabled interfaces

Different devices can communicate with each other and exchange data via the various interfaces. Communication between devices often involves sensitive data, which could cause immense damage if it were to become known or be manipulated. Significant damage can also be caused due to inadequately protected configuration interfaces.

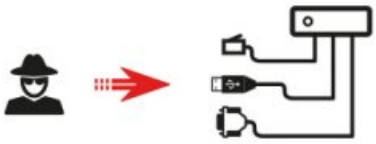

Figure 2.29: Module ES/EXTERNAL

The ES/EXTERNAL sub-module checks the protection level of these cabled interfaces. The device is not opened. Only the interfaces that are externally accessible are tested. These include, for example, Ethernet, serial ports or optical sensors. Other interfaces, such as proprietary interfaces, can also be tested. This module is primarily suited to initial security considerations and can also be combined optimally with the ES/INTERNAL (see Section 2.11.4) and ES/PROTOCOL (see Section 2.11.5) modules for a deeper analysis of the products or individual interfaces.

Note: If only the web application or a web service offered by the test device (e.g.: configuration side of a router) is to be tested, the WEBAPP (see Section 2.2 on Page 25) or WEBSERVICE (see Section 2.3 on Page 29) modules are ideal.

Questions

During the penetration test, SySS answers the following questions, amongst others:

- Checking the encryption: Is the transferred data adequately encrypted? Can the encryption be broken or bypassed?
- Are there weaknesses in the authorization and authentication system?
- Manipulation of the transferred data: Is it possible to assume a machine-in-the-middle position in order to record or manipulate sensitive data?
- Replay attacks: Can recorded sequences be replayed to trigger an action in the device? (Example: Deactivation of locking or alarm systems)
- Can sensitive data be extracted through misuse of the interfaces?
- Can a device be composed via the existing interfaces?

2.11.3 ES/FIRMWARE: Security analysis of firmware

There are barely any electronic products these days that work without corresponding firmware. In the same way that the products differ, the features of the implemented firmware, which is often aligned with the respective use case of the product, also differ. This applies, for example, to an adapted Linux system, barebone firmware that communicates directly with hardware components without abstraction, or a real-time operating system (RTOS). The wide range of possibilities often makes it very difficult to design the firmware securely. Issues such as the design of a secure update mechanism "Over-the-Air" (OTA) or the problem of a secure licensing procedure can also pose challenges.

Figure 2.30: Module ES/FIRMWARE

The security test of the ES/FIRMWARE module aims to analyze these very areas.

In most cases, static key analysis or reverse engineering are performed in this module. For this reason, SySS recommends providing the firmware as a file. If the firmware is encrypted, an unencrypted version should also

be provided. Although only the device software is tested, SySS still recommends supplying the corresponding hardware. Only in this way can any weaknesses that are ascertained be fully traced and tested.

Questions

During the penetration test, SySS answers the following questions, amongst others:

– What type of firmware is used? (Linux-Image, binary image format, etc.)
– Do the encryption and signing of the update files work reliably?
– How does the update process work?
– How do the authorization and authentication processes work?
– Are there weaknesses in the services provided? (Remote Code Execution, SQL Injection etc.)
– Can secret code be extracted, which can be used for further attacks (such as against the cloud infrastructure)?

2.11.4 ES/INTERNAL: Security analysis of internal interfaces and memory components

Sensitive information, such as credentials for a back-end system, client certificates or passwords for maintenance access, is often stored on modern devices. Internal functionalities – such as algorithms that process sensor data – are also often regarded as company secrets. Since the devices have not been normally subjected to a physical inspection by the vendor, third parties can use targeted attacks to try and extract stored data or manipulate the device in another way.

Figure 2.31: Module ES/INTERNAL

The ES/INTERNAL sub-module therefore focuses on attack vectors in an attack with physical access to the test object. SySS analyzes the possibilities of attacking internal data memory (e.g. eMMC, Flash-ROM), interfaces (e.g. UART, I²C, JTAG, CAN) and the components used (e.g. controllers, Crypto-IC). For this, it is necessary to open and/or dismantle a device. Therefore, it may be the case that additional lines are attached to the PCB or that individual chips, such as memory modules, are tested when desoldered and released from the product. Provision of the product in several versions is therefore urgently recommended.

Questions

During the penetration test, SySS answers the following questions, amongst others:

– Are there effective ways of protecting against extraction of the firmware?
– Are credentials for the bootloader or maintenance interfaces well-known or easily guessed?
– Can the content of memory modules be extracted through the attachment of lines or the chip-off method?
– Can credentials or private code be extracted? (These can be used for further attacks, e. g. against a Cloud back end.)
– Bus snooping: Unprotected interfaces on the PCB can be bugged during an attack. Can secrets be read out in this way or can subsequent encryption be bypassed?
– Lack of protection against manipulation (tampering): Can the way in which devices work be changed or can secrets be obtained without this being noticed?

2.11.5 ES/PROTOCOL: Security analysis of protocols

A device communicates with other systems using different protocols. In the process, data is exchanged or commands are sent. Since it is often sensitive data that is involved, whose manipulation can cause significant damage, the security mechanisms of these protocols are of particular importance.

Figure 2.32: Module ES/PROTOCOL

The protocols used here, which are often proprietary, are investigated in more detail in this module. Typical aims of the test are to test the encryption strength and to protect the integrity of the transmitted data. Testing for logic errors, which can often lead to security incidents, is also part of the penetration test. When using complex protocols, it is also possible to check whether authorization and authentication security mechanisms are bypassed.

Note: In contrast to the ES/EXTERNAL sub-module (see Section 2.11.2), with which the possibility of compromise via externally accessible interfaces is tested, the focus of this sub-module is on a deeper investigation of a specific protocol.

Questions

During the penetration test, SySS answers the following questions, amongst others:

- How strong is the protocol encryption? Can it be canceled or can sensitive data be extracted?
- Can data be manipulated?
- Can the devices be made to behave undesirably through logic errors?
- Replay attacks: Can recorded sequences be replayed to trigger an action in the device? (Example: Deactivation of locking or alarm systems)

2.11.6 ES/WIRELESS: Security analysis of wireless interfaces

Different devices can use different wireless technologies to communicate with each other, to exchange data or to send each other control signals. It is hard to imagine life without wireless connections either in private life or in an industrial environment, whether it's the networking of large industrial plants, building access controls or even contactless payment or the use of wireless headphones. The corresponding data is often worthy of protection and manipulation of this data can cause significant damage. Moreover, acquisition of control can occasionally put people's safety at risk. Therefore, the security of these communication channels is particularly important. In particular, it is of considerable importance that an attack requires only physical proximity but not actual physical access. Devices can even be completely compromised in this way.

Figure 2.33: Module ES/WIRELESS

The security test of the ES/WIRELESS module aims to analyze these wireless technologies and the communication protocols used. SySS analyzes the attack possibilities, including via Bluetooth/BLE, Zigbee, LoRaWAN,

ZWave, Wi-Fi or even NFC/RFID. Analysis of data that is sent via mobile communications (e.g. LTE) is also feasible. In addition to the implementation of protocol stacks, another main focus is the search for configuration errors in the respective technology used.

Note: The analysis of other wireless protocols, including proprietary protocols, is potentially also possible, given that SySS is provided with the necessary hardware and software. We would be pleased to offer advice in this regard. To test Wi-Fi infrastructures in the corporate environment, the Wi-Fi module (see Section 2.8 on Page 51) should be used.

The ES/WIRELESS sub-module can, in principle, be implemented without opening the device, using a black box approach (see Subsection 1.1.3 on Page 8). However, it is recommended that a wireless-based penetration test be implemented at least using a gray box approach (again, see Subsection 1.1.3) (provision of documentation, specifications, etc.), as this provides a more efficient and economical result, according to experience. The way in which the protocols work and possible test scenarios can be discussed in advance in a joint workshop. For a deeper analysis, the ES/INTERNAL (see Section 2.11.4) and ES/PROTOCOL (see Section 2.11.5) modules should also be considered.

Questions

Example questions, which can be answered during the penetration test, are listed below for different technologies:

Bluetooth Smart (BLE)

- Can the communication be read in plaintext?
- Can characteristics/services be read/written without authentication?
- Which data is transferred via BLE and can it be manipulated?
- Which security mode is used?
- Are BLE machine-in-the-middle attacks possible, e.g. through spoofing of the BLE-MAC address?
  Tool selection: Sniffle, nRF Sniffer for Bluetooth LE, nRF52 Dongle, nRF52 DK

RFID and NFC:

- Can boards be cloned or emulated?
- Can credentials be manipulated?
- Is the saved data adequately secured?
- Can sensitive data be extracted?
  Tool selection: Proxmark3, card reader, NFC-enabled smartphone

Wi-Fi

- How is the defined Wi-Fi protected (authentication)?
- Are current protective measures active (e.g. protected management frames)?
- Are subscribers isolated from each other or should they be?
- Are there configurative weaknesses (e.g. password repetition, weak passwords, etc.)?

ZIGBEE

- Are packets correctly encrypted?
- Is there encryption at network or application level?
- How are the codes exchanged?
- How are new devices programmed (is there a Master Key)?

2.12 OT: Operational technology security

Summary

Operational technology (OT) refers to the hardware and software used to control and monitor machines, production systems and industrial processes. The security of the OT is essential for a company's operations and business continuity. Attacks and failures can have fatal consequences, e.g. for the continuation of production, the logistics of raw materials and products, the supply of energy, water and operating materials, or for the functionality of the machines used. Depending on the industry, an incident may also pose a risk to the life of employees or the general public.

The special requirements for availability, functional safety and resilience to cyber attacks pose a challenge. The very long service life of the devices used and the heterogeneous architecture of the OT landscape, which requires cooperation with various manufacturers and service providers, often make it difficult to develop a comprehensive security concept. In order not to disrupt operations, it is difficult to carry out penetration tests on the productive infrastructure.

SySS offers several services that are tailored to different needs and requirements of OT.

Figure 2.34: Module OT

2.12.1 OT/WORKSHOP: Workshop on OT environments

Summary

The OT infrastructure is evaluated with respect to possible weaknesses and vulnerabilities in a workshop. The project result is a report which contains recommendations on mitigating identified risks as well as suitable pentest scenarios tailored to the individual environment.

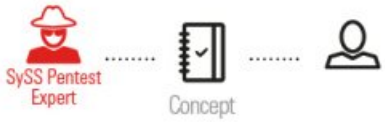

Figure 2.35: Module OT/WORKSHOP

Starting situation

In order to carry out an analysis and risk assessment of the OT infrastructure, the architecture and the systems used must first be known and a security concept must be developed. The OT workshop is suitable both for creating an overview of the system landscape with simultaneous security analysis and for evaluating an existing concept or planned changes as part of a project to improve security. The workshop is also suitable for preparing a penetration test and the practical development of suitable test scenarios.

As the OT workshop does not require any intervention in the systems, it can be carried out independently of maintenance windows and there is no risk to operations.

Objective

The SySS OT workshop uncovers weaknesses and potential for security improvements in the OT environment. It is therefore the starting point for the development and implementation of further consulting services and technical analyses. The workshop itself is the first step towards raising awareness of threat scenarios and protective measures in the OT network.

In collaboration with the customer, the consultant team carrying out the workshop gains an overview of the current status of the OT environment and derives a customized list of current problem areas and a catalog of appropriate measures. SySS examines the existing systems and concepts from an attacker's perspective in order to uncover potential threats to the operation and security of the processed data. SySS explains possible attacks in order to provide employees with an effective view of risks related to OT security.

Following the workshop, the information covered is prepared and made available in the form of a report. The project includes a documentation module for this purpose (remotely at SySS).

As part of the workshop, SySS works together with the customer to develop appropriate penetration test projects in the OT environment, which are outlined in the report with reference to the existing infrastructure. This can be, for example, a test of the transitions between IT and OT, a pentest of the engineering workstations and OT clients used, or an analysis of the process control system.

Implementation

The workshop usually takes two days on-site and ideally includes an inspection of the facilities. The documentation is then created remotely.

After an introduction of participants and an overview of the existing system landscape, a selection of the following topics is covered:

- Assessment of the protection requirements and criticality of the OT systems
- Overview of typical technical threat scenarios
- Analysis of existing security measures
- Network structure with a special focus on separation and connections between networks
- Production control system and its interfaces
- Authentication and identity management
- Remote access solutions
- Patch and update processes
- Data storage and backup strategy
- Physical security
- Service providers
- Wireless networks (e.g. Bluetooth, WLAN, LTE, etc.)
- Mobile devices (e.g. laptops, handheld scanners, etc.)
- Virtualization
- Logging and monitoring
- SIEM and incident response processes
- Autonomous operation of machines

The focus of the workshop is always tailored to the customer's needs. In this regard, the workshop agenda is discussed and defined in the kick-off meeting. The depth of the individual topics is adjusted during the workshop as required. During the workshop, it is possible to raise further questions.

It is advisable to make existing documentation and concepts available to SySS in advance. During the workshop, it is also possible to examine and discuss which security analyses and penetration tests make sense in a specific

case and can be carried out in the future. It is also possible to use parts of the workshop for initial practical security checks.

Cooperation by the customer

Project preparation: Before the workshop begins, a selection and prioritization of the topics to be covered should be defined in order to tailor the workshop ideally to the customer's needs. As it makes sense to inspect the facilities, it should be clarified what preparations are necessary to allow the SySS consultants access to the company's premises. Necessary personal protective equipment is also discussed in advance.

Contact person: The workshop is conducted as a discussion with the appropriate contact persons for the individual topics. Depending on the structure of the company, these may be different people (in which case the topics should ideally be grouped together in the kick-off meeting so that the time blocks can be used efficiently).

Dependencies: Conducting the workshop does not require the systems to be shut down. Technical inspections only take place in close collaboration. The availability of the production facilities always has top priority.

Tip by Sebastian Schreiber

Use our OT workshop to raise the security awareness of your employees in the operational technology. Knowing the attacker's perspective is worth its weight in gold in everyday life to nip security problems in the bud.

2.12.2 OT/PENTEST: Security testing of OT environments

Summary

The systems and networks of the OT environment are analyzed with respect to vulnerabilities which, e.g., could allow an attacker to escalate privileges or gain access to secured areas. High priority is given not to disrupt production activities.

Figure 2.36: Module OT/PENTEST

Starting situation

Process control systems often use typical IT components such as an Active Directory domain for managing user accounts, virtualization servers, file servers, Windows clients and database servers. In order to benefit from digitalization, data is exchanged with the company's IT systems. Furthermore, remote access from the IT network or by service providers is usually necessary for the operation of OT. An attacker succeeding in penetrating the OT network poses a high risk to the OT components.

Objective

The aim of penetration testing in the OT environment is to examine the systems for practically exploitable vulnerabilities and security risks. Depending on the desired scenario, either an attacker located in the OT network is simulated or the perspective of a compromised computer or an internal perpetrator is assumed by providing SySS with low-privileged credentials. This examines whether it is possible to move laterally to other systems and network areas or to escalate permissions. Communication paths – for example with the internet – are also examined.

The availability of the systems has top priority.

The systems of the OT infrastructure can be divided into levels with increasing abstraction of the physical processes using the Purdue reference model[3]. A pentest can be carried out at different levels. In zone 3, the process control system with the components used, such as engineering workstations or clients in the OT network, can be checked. In zone 2, the isolation of individual control systems and the security of operating units can be tested. In the OT DMZ, it makes sense to check the separation between OT and company IT as well as the security of the services offered. Jump servers and the remote access used should also be checked for security.

Implementation

The implementation is based on the LAN/CLEAN (see Subsection 2.4.1 on Page 34) and LAN/TRAINEE (see Subsection 2.4.2 on Page 34) modules, taking into account the special requirements for availability in the OT network. Automated vulnerability scans are only carried out here in a targeted manner and in close consultation. Depending on the devices used, port scans can also be carried out at a lower speed only.

If a test environment is available, it should be preferred to the production environment. Here, practical checks of possible vulnerabilities can be performed with a much lower risk, and the testing depth can therefore be increased.

SySS produces detailed documentation on the vulnerabilities found, including recommendations on how to rectify them. A management summary is also included.

Cooperation by the customer

Preparation: Ideally, SySS receives network plans as well as the specification of the machines and devices used in advance of the project. On the one hand, this allows the pentest to be carried out efficiently within the available test time. On the other hand, it minimizes the risk of unintentional issues to system availability. As the pentest usually takes place on-site, it should be clarified what preparations are necessary for the SySS consultants to gain access to the company premises. The necessary personal protective equipment is also discussed in advance.

Contact person: During the test, it is essential to monitor the smooth operation of the systems. In addition, a contact person must be available who can take the right steps to resume operations in the event of any problems.

Dependencies: If the test is carried out during productive operation, there is a risk that operation will be disrupted by the pentest. If maintenance windows or repair weeks are available, these are ideal for carrying out a pentest. Likewise, if available, a comparable test environment can be assessed instead of the productive infrastructure.

[3] https://www.energy.gov/sites/default/files/2022-10/Infra_Topic_Paper_4-14_FINAL.pdf

Tip by Sebastian Schreiber

An ideal time for a penetration test of the OT network is the delivery of a new machine or a new process control system. As part of a Site Acceptance Test (SAT), cyber security can also be tested and existing defects can be rectified by the manufacturer before commissioning.

2.12.3 OT/ANALYSIS: Security analysis of OT components

Summary

An OT component (e.g. a programmable logic controller) is checked for vulnerabilities by analyzing its available interfaces in the SySS lab. Due to the execution in a test environment, there is no disruption to production activities.

Figure 2.37: Module OT/ANALYSIS

Starting situation

Industrial control systems are often very sensitive to faulty or defective data. Testing during productive operation therefore carries a high risk of impairing the availability of the components. Nevertheless, potential vulnerabilities in the OT devices used should be identified in order to reduce the risk of attacks.

The availability of updates for OT components is often limited. Even if updates are available, it often takes a comparatively long time for them to be installed due to maintenance contracts, certifications or maintenance windows. A pentest of the components used can help to assess the realistic threat situation and thus facilitate the prioritization of updates.

Devices such as programmable logic controllers (PLC), remote terminal units (RTU), remote access solutions (RAS) or human machine interfaces (HMI) are suitable for analysis. Ideally, the device is available in a test environment and can be mailed to the SySS lab for analysis.

Objective

The pentest of the OT component typically has the following objectives:

- Identification of all interfaces and services offered, including their separation from each other
- Analysis of authentication and authorization
- Examination of resilience to unexpected data
- Assessment of data transmission, including the encryption used and possible manipulation
- Audit of the update mechanism
- Analysis whether it is possible to extract sensitive data or compromise the system

Where possible, desired testing scenarios as needed by the customer are accommodated.

Implementation

The implementation is based on the EMBEDDED modules. Depending on the specific device and its available interfaces, the communication protocols, administration and remote access as well as the update process are analyzed. The device is examined in the SySS lab in Tübingen. Depending on the test scenario and the focus of the analysis, the external interfaces, the services offered on the network side or the internally installed components are tested. The device can also be opened and components can be desoldered for this purpose.

Any weak points found and recommended improvements are presented in a detailed report. This also contains a management summary.

Cooperation by the customer

Preparation: Ideally, the OT components are analyzed in the SySS lab. Therefore, the test components must be sent to SySS in advance of the project, e.g. by mail. They are returned after the project. Documentation and – if available – the specification of the connections should be made available to SySS.

Contact person: During the project period, a contact person should be available remotely for questions regarding the use of the OT device in the specific OT environment.

Dependencies: As the OT device is tested in the SySS lab, no productively used device can be used. On the one hand, the device would otherwise not be available for operation for the duration of the project, and on the other hand, there is a risk that irreversible changes could be made to the device during the test. Ideally, a replacement component or a test device is used that is identical in design and function to the productively used device.

Tip by Sebastian Schreiber

When it comes to the security of OT components, do not trust the information provided by the manufacturer. Holding a device in your hand (and opening it up if necessary) is the best way to uncover actual security flaws and potential attacks.

2.13 SOFTWARE: Security analysis of software solutions

Summary

Software components and products are examined for vulnerabilities during this security test. The focal points of the security analysis are security-related functions such as authentication, authorization and encryption. Using various analysis methods, a search is carried out for possible security vulnerabilities in the software to be tested. These vulnerabilities can be exploited by attackers in different ways. Well-known examples of software vulnerabilities include errors during processing of user inputs which can be misused, for example, to execute any program codes or errors in the privilege concept that permit unauthorized access to functions or data.

Figure 2.38: Module SOFTWARE

Starting situation

Software forms an integral part of modern IT systems and processes. Software products and individual software components are firstly very important for the proper performance of workflows and business processes, and also for ensuring information security. Security-related functions such as authentication, authorization and encryption represent in this case important elements which should not contain any security vulnerabilities.

Objective

By agreement, security-related functions of the software product to be tested are analyzed for vulnerabilities during the security test. In this case, it is verified whether defined protection objectives such as confidentiality, availability and integrity may be at risk.

Depending on the software product, the objectives of a possible attack may be, for example, to gain unauthorized access to available functions or data, carry out privilege escalation in the target system via the installed software product, circumvent implemented protective mechanisms such as digital rights management or steal intellectual property relating to the functionality of the software product.

Implementation

If possible, the security analysis of software products takes place in the SySS laboratory in a suitable infrastructure. Depending on the technologies used for the software product, e.g. programming languages and runtime environments, as well as demands on the target platform, e.g. the processor architecture and operating system, different tools and analysis methods are used.

Unlike open-source software, the operation of many software products is not immediately apparent due to the lack of access to the source code (closed-source products). Therefore, various reverse code engineering methods are used for the vulnerability analysis of software products which are only available in compiled form. On the one hand, this includes the static code analysis of binary programs using software tools, such as decompilers (e.g. ILSpy for .NET applications or JD-GUI for Java applications) and disassemblers (e.g. IDA Pro or Hopper for different executable file formats from different platforms). On the other hand, this also covers the dynamic code analysis using software tools such as debuggers (e.g. OllyDbg, x64dbg, dnSpy or GNU debugger) and dynamic binary instrumentation tools (e.g. Frida or DynamoRIO).

If all or some of the source code can be provided for the software to be tested, this is urgently recommended in order to reduce the test expenditure and improve the test performance of the security analysis.

Cooperation by the customer

Test preparation: In order to carry out a security test of a software component or a software product, SySS must be provided with the corresponding software either in an executable form for the test in the SySS laboratory or by means of suitable access to a test instance. In order to attain the best possible test results, there should be no restrictions on the use of the software product. In an ideal scenario, SySS has complete control over the test system containing the software forming the subject of the test. In the case of white box tests of software products, SySS should be supplied with the source code of the software to be tested together with documents such as manuals and technical documentation.

Contact person: The contact persons responsible for the security test of a software product should be available during the test period.

Dependencies: SySS must be notified of organizational and technical dependencies. This can occur during the kick-off discussion. If the software product or software component to be tested is, for example, inoperable in isolation and cannot therefore be tested separately from other systems, SySS must be told which tests can be carried out on dependent systems and which contact persons are available in this case.

Tips by Sebastian Schreiber

Estimate the time required for the security test of a software product very generously! Check what dependencies, communication relationships and positions of trust of the software product exist in relation to other systems. A full test of these systems is normally beneficial if they already form an organizational and technical unit. Then select the suitable accompanying test modules for the test.

2.14 Other modules

In addition to the above-mentioned classic test modules, some other focused test scenarios have proved worthwhile in the last few years. They will be described in the following sections.

2.14.1 RECON: Inventory of the attack surface

Summary

Depending on the test perspective, SySS identifies the customer's attack surface that can be seen by a third party. This attack surface may include, for example, the IP addresses and IP address areas which can be directly accessed from the internet and traced by means of publicly available information, as well as web applications and services. An analysis of the network areas accessible from a certain starting position can also produce important results within company networks. This test module supports inventory measures and the selection of systems to be tested for other modules (e.g. IP-RANGE, WEBAPP, LAN and PIVOT).

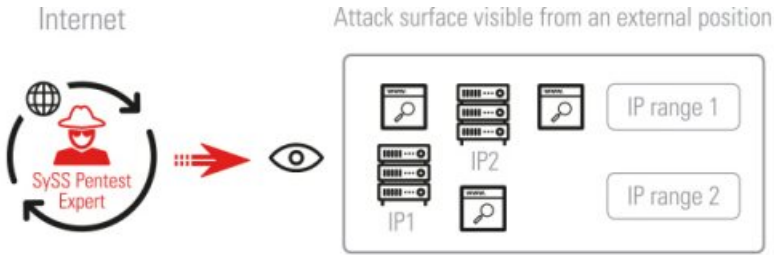

Figure 2.39: Module RECON

Starting situation

In particular, large companies with diversified IT departments often do not have an extensive overall picture of their own IT infrastructure right down to the last detail. Different areas, e.g. other departments or other organizational units, other locations or subsidiaries frequently work without any agreement. This may be one of the reasons why "shadow IT" is growing inside the company. Shadow IT means that unrecorded systems, i.e. those which are administered through automated patch management and which can be accessed via the internet, exist in the company's own network and may pose a potential security risk.

In contrast, smaller companies face the problem of having fully outsourced their own IT management. They therefore lack an overview of their own gateways which may exist.

In these cases, but also whenever a correctly implemented asset management system exists, it may be interesting to obtain a realistic picture of a company's own visible attack surface from different angles.

Objective

Security tests are only normally carried out on IP addresses and services which were selected and stipulated beforehand by the customer – with the support of SySS. If, especially in large internationally distributed customer networks, this selection cannot be made or if the objects to be tested first have to be identified, an inventory can be carried out as provided for in this module.

The objective is therefore to produce an overview of, for example, systems (IP addresses or IP address areas) or web applications and services which can be explicitly assigned to the customer and are visible from a certain position of the attacker. Typical implementation forms of this module are:

- Identification of the public IP address areas to be assigned to the company
- Determination of the systems which are accessible from the internet (perimeter)
- Identification of the web applications and services published in the internet by the customer
- Compilation of the customer's e-mail addresses or employee data that has been published on the internet
- Identification of active systems and their accessibility in selected internal network areas

Any errors in assignment (e.g. incorrect Regional Internet Registry (RIR) inputs) can also be detected and, if necessary, candidates for a subsequent security test can be selected here. These candidates are selected after verification by the customer himself. The customer is also able in this case to correct errors in his own documentation or instruct service providers (e.g. ISPs) to take this action. Due to general legal conditions, SySS cannot act independently in this case since it must always be ensured that third parties are not adversely affected.

Implementation

Different procedures are used depending on the test scenario. During the conventional method of identifying the external attack surface, public sources such as RIR databases (in Europe, RIPE) or DNS are scanned based on already known information (e.g. domain names or hostnames, e-mail addresses, etc.). Mail routing by the customer and content from websites, which may provide information on company links, are also taken into account in this case.

With inventories inside company networks, SySS carries out a scan from network areas stipulated by the customer to determine what other network areas are accessible and what systems are active in these network areas. This can also be ideally combined with an isolation analysis (technical verification of the firewall regulations). Different port scans represent the focal point of the test activities in this case.

Cooperation by the customer

Test preparation: Depending on expectations, already known information such as IP address areas or domain names/hostnames can be reported to SySS. This can certainly speed up other research. Otherwise, however, the black box perspective can also be adopted.

Contact person: In order to make a distinction between the customer's systems and third-party systems and coordinate results with the customer's own documentation, a contact person should also be available in this module.

Tips by Sebastian Schreiber

Specify your exact expectations for the project during the kick-off discussion! When determining the external attack surface, it is standard procedure for our consultant to only look for IP address information and (sub)domain names, for instance. If you are also interested in seeing which e-mail addresses can be found, for example, from the internet, always inform the consultant accordingly before the start of the project!

2.14.2 SOCIAL: Social engineering

Summary

When stealing data, attackers use all possible means of obtaining valuable information. Whereas it has been normal practice for many years to use technology to protect hardware, software, applications and networks against attackers, attacks are now increasingly taking place on an interpersonal level and are called "social engineering" (SE). The objectives of this module are to help your employees to protect themselves better against manipulation attempts, as well as to increase their awareness of how brazen and unscrupulous attackers act at times and of how attackers have no inhibitions in gaining access to data and company premises through the use of lies and false information. The module helps you to record whether your already implemented measures against social engineering are working and where you have a need for readjustment. Social engineering is especially insidious because it plays on basic human values and our upbringing as social people respecting others, and perverts these values. Very careful action is therefore required in this case.

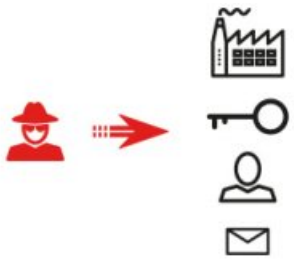

Figure 2.40: Module SOCIAL

Starting situation

"I have gained unauthorized access to some of the world's largest companies and successfully hacked into some of the most intractable computer systems that have ever been developed. I used technical and non-technical methods in this case in order to acquire the source code of different operating systems and telecommunications equipment so that I could study their vulnerabilities and internal modes of operation." (Kevin Mitnick: The Art of Deception)

As so aptly described by Kevin Mitnick, "modern" hackers no longer only use technical tools to access companies they want to attack. There has been an increasing number of social engineering attacks, in which employees – in other words, the human factor – play a key role. These types of attacks are often very successful. In order to also provide you with an in-depth analysis of your security against these attacks, SySS offers its own social engineering module in addition to current technology-based tests. Social engineering can be carried out either independently of other tests or in combination with other modules.

Objective

Even though the focal points of social engineering attacks are people and their weaknesses, the objective of a social engineering test is never to embarrass or discredit individual employees. In these tests, it is more a question of testing awareness measures, analyzing and improving processes, and increasing employees' awareness of this type of attack. Social engineering techniques are also used in modules such as Physical Pentest (see Subsection 2.14.3 on the next page) or Red Teaming (see Chapter 3 on Page 89) if no there is no other available option to attain the objective set by the customer.

The objective of the test is to answer the following questions:

- How aware are your employees?
- Do awareness measures work?
- Are the processes that are supposed to make these attacks more difficult or prevent them known, and are they implemented in practice?
- Do the technical precautions work?
- Are there gaps in the procedure in the event of suspicion?

Implementation

Social engineering attacks are carried out solely by specially trained employees of SySS who have received specialized training and been made aware of legal and ethical aspects (see Section 3.3 on Page 93). They always handle the topic very responsibly and cautiously. The utilized techniques are always adapted to the respective module and the individual situation. Generally speaking, use is made of the following non-violent techniques which employees must expect to encounter during their normal daily work:

- **Phishing and spear phishing e-mails:** Employees are contacted by e-mail and requested to carry out a specific act (frequently entering access data on a website).
- **Pre-texting:** An attacker attempts to invent a scenario which legitimizes an intended act, for example to gain unauthorized access to a building or a site, or induce an employee to disclose specific information.
- **Phone calls and text messages:** As well as e-mail, employees may also be contacted by phone or by text message. Publicly accessible contact data is utilized in this respect. This technique can also be used to answer calls made by employees on account of incorrect information during conversations or in e-mail signatures.
- **Sending letters:** On rare occasions, a process or a guideline stipulates that communication must take place by post. For this reason, sending letters may also form part of this test. In this case, signatures may have to be copied. However, this only takes place with the written approval of the person in question.
- **Search for persons using public business community profiles:** Background information is essential in order to create the most authentic scenario during pre-texting. For this purpose, SySS primarily uses publicly accessible information in the internet and any network platforms. However, print media, radio or TV reports may also contain interesting facts about a company or individual employees.
- **On-site techniques:** If the selected test module stipulates that the test is also carried out on-site, other techniques such as stealing unattended security tokens, copying employee IDs, using a disguise and presenting a false identity, or installing remote accesses in the company network may be used in this respect.

When the test is performed, it is always ensured that the privacy of the customer's employee is protected. For this reason, SySS does not mention the name of affected employees during personal discussions or in the final report.

Cooperation by the customer

Since the social engineering module is a very complex test module, it is vital to prepare for and follow up after the test. During the preparations for this test, the general conditions along with the utilized techniques are defined in a joint workshop. All employees who may be part of a social engineering test must also be informed during the preparations that tests of this kind are carried out if SE tests are not already part of corporate culture. During the test, an employee must always be available as a contact person for SySS so that critical situations can be resolved quickly and easily. The focal point here is always the protection of affected employees. Past experience shows that handling of the test results with SE modules is always more complex than with customary technical penetration tests. However, the following principle applies: Even if a specific employee was attacked, the problem here is not the ignored/undefined process or the lack of preparation or training of employees. As defined in SE ethics (see Section 3.3 on Page 93), neither their name nor other internal details are mentioned in the final report.

Tips by Sebastian Schreiber

Prepare your employees for social engineering tests by regularly announcing them and stipulating specific periods. This will increase acceptance of these tests and prevent dissatisfaction and irritation. You will also create awareness of real attacks. Before the test is carried out, check whether the employees concerned have sufficient knowledge to handle unique situations, and create processes which offer your employees good support in the event of an emergency.

2.14.3 PHYSICAL: Physical pentest

Summary

Worthwhile attack targets include not only web applications, computers or networks which can be accessed from a more or less large distance. There is also a wealth of data which can primarily be stolen when an attacker gains access to buildings and is physically present. During a physical pentest, SySS examines all possibilities of entering a building without authorization, accessing devices and, in this way, capturing data.

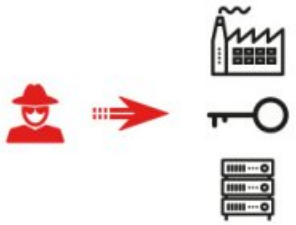

Figure 2.41: Module PHYSICAL

Starting situation

In every company building, there is a risk that unauthorized persons could gain access to the site, building and/or rooms and equipment (e.g. research and development areas of an industrial company). Consequences:

– Property may be damaged or stolen
– People in the building may be exposed to danger

– Technical systems may be manipulated
– Data may be obtained more easily through direct access to computer systems, photocopiers, mobile devices and, not least, the network infrastructure

In accordance with Article 5 of the General Data Protection Regulation (GDPR), controllers of personal data must have implemented technical and organizational measures to protect data against access by unauthorized persons.

Objective

The objective of a physical pentest is to verify the security of a building. This type of test will help to detect threats, assess their likelihood of occurrence and damage potential, and evaluate the risk which these threats pose to the organization. In addition to technical precautions, tests are carried out on the access processes, access control processes and monitoring processes. The following questions are answered through the test: Are the security concepts adequate? Are there gaps in the security concept? Do the technical precautions work? Are the stipulated procedures observed? Do the technical and organizational measures ensure that personal data is adequately protected?

Implementation

The consultant attempts different methods to gain access to the building or the agreed rooms. For this purpose, he/she initially obtains different information which is also available to an attacker (information on the internet, observations, etc.). Physical pentests normally contain the following procedure:

– Acquisition of information from public sources
– Observation of the building, site and surroundings
– Analysis of access options
– Identification of access controls
– Observation of authentication measures for employees and guests
– Analysis of effectiveness of access controls
– Search for options to circumvent the installed protective measures

Social engineering methods are also used during the tests (see Subsection 2.14.2 on Page 79). For example, a consultant might try to get into the building by means of tailgating, i.e. by simply following an employee who has just opened the door. Unplanned maintenance work or a visit personally announced by the attacker – from an external telephone number with a fake internal name – would also be possible. When implementing the tests, we always draw the line at methods whose objective is to place employees under severe stress, simulate emergency situations or something similar. We therefore only use specially trained consultants for these tests. Our ethical principles for social engineering, as described in Section 3.3 on Page 93, always form the basis of our work.

Cooperation by the customer

During an active test of access controls, an attempt is made to circumvent physical security measures. This can certainly be regarded as a break-in. It is therefore extremely important here that you give us a detailed explanation of the circumstances under which the test is to take place. Depending on the security of the building, it may be necessary to give our consultants "carte blanche" to ensure, for example, that the police are not called in response to an unsuccessful or detected penetration attempt. In divided buildings or special situations, additional information may be required. For example, it must be defined which areas should not be penetrated

due to security reasons and where no such attempt should be started either. Just like in all tests using social engineering methods, a discussion is necessary in the run-up to a test, during which we clarify permitted and excluded methods, the need to raise awareness among employees beforehand and afterwards, and other similar aspects. Past experience shows that handling of the test results with this module and other social engineering modules is always more complex than with customary technical penetration tests. Even if a specific employee was attacked, the problem here is the ignored/undefined process or the lack of preparation or training of employees.

Tips by Sebastian Schreiber

Prepare your employees for the physical pentest and other social engineering tests. You prevent dissatisfaction and irritation concerning the test and simultaneously create awareness of genuine attacks. Nobody wants to fail a test. An attentive employee is always helpful when it comes to preventing abuse and the resulting damages! Before the test is carried out, check whether the affected employees are sufficiently well-informed to handle special situations. Make sure that there is a procedural instruction or a process for dealing with unauthorized persons on the company's premises.

2.14.4 PIVOT: Compromised demilitarized zone (DMZ)

Summary

This module is based on the scenario that an attacker manages to take over at least one system in the demilitarized zone (DMZ), such as a web server. SySS will attempt, for example, to penetrate further into the DMZ or internal network segments, or to show what information an attacker can obtain in the DMZ by exploiting positions of trust or other vulnerabilities.

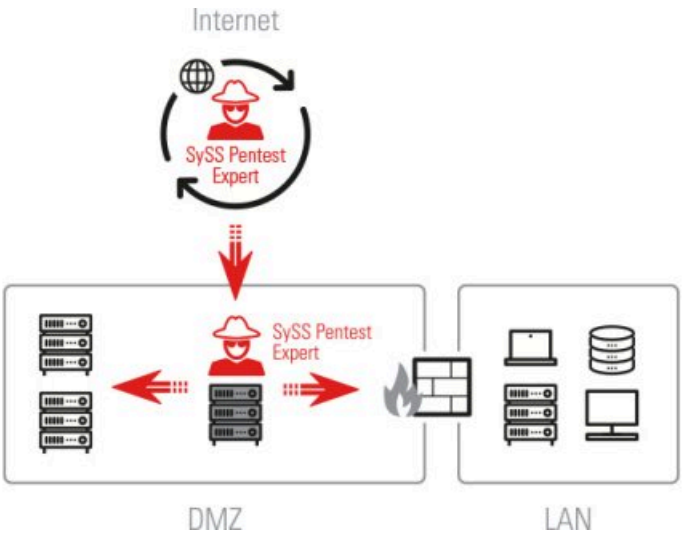

Figure 2.42: Module PIVOT

Starting situation

Often the target of attackers is to copy information from database management systems connected to web applications by exploiting vulnerabilities such as SQL injection. In many cases, the attackers are interested in data such as passwords, e-mail addresses or payment details stored in the database so that they can be sold at a profit.

If, however, the intention of attackers is to deliberately cause harm to a specific company, they only use these vulnerabilities as an entry point for further attack activities and the affected servers are used as "pivot systems". The actual motivation of the attackers is normally to penetrate the internal company network in order to misappropriate or steal critical company data, e.g. intellectual property. In a worst case scenario, this attack forms part of an advanced persistent threat (APT) in which other attack techniques such as social engineering or phishing methods are used.

Objective

Within the framework of this module, SySS will analyze which possibilities are available to attackers who were able to compromise a server in the DMZ during a successful attack. This system is then used as a pivot in order to redirect deeper attacks in this way. A pivot can, for example, exploit the positions of trust between this server and other servers in order to carry out attacks on systems which are not directly accessible from the internet. The objective is to penetrate more deeply into the DMZ or internal network areas.

Implementation

As it is not possible to assess whether SySS will manage to penetrate the customer's DMZ under its own steam, for example as part of a web application analysis (see module WEBAPP in Section 2.2 on Page 25), SySS is normally provided access to a server in the DMZ. This also makes the PIVOT module independent of other test modules. The procedure is based roughly on the following pattern:

– Uploading of the necessary tools (e.g. a payload with proxy/VPN capability) to the pivot system
– Analysis of the pivot system (e.g. privileges, local credentials, network interfaces, open links, etc.)
– Pivoting (often referred to as "island hopping"), which encompasses testing the accessibility of other systems within the DMZ and in internal network areas, as well analyzing other systems

If required, the following tests can also be performed:

– Deliberate attempt to attack certain internal systems (e.g. file server or mail server)
– Penetration test for dedicated network areas (technical verification of the firewall policy)

Cooperation by the customer

Test preparation: In order to perform the test, SySS must always have access to a system in the DMZ. This system is normally a web server or an application server. Database servers, which are often located in separate demilitarized zones, can also be used for this purpose. In order not to disturb productive operation, a technically identical clone of such a system can be ideally used. It is important for the system to have a configuration which is as identical as possible to the productive instance. SySS therefore has all attack possibilities which could also be available to a real attacker.

Access is ideally established via SSH. SySS should be granted the same privileges which are also held, for example, by the service account of a web server or an application server. The context here is that an attacker very probably holds these privileges if they managed to successfully exploit a vulnerability in a web application.

Alternatively, special VPN access (end-to-end) can also be set up or protocols such as RDP or VNC can be used.

Contact person: In the previous module descriptions, some reasons have already been mentioned as to why (technical) contact persons and their availability are very important during the test period. These reasons also apply, in particular, to this module. During the test, contact persons can answer any questions which are used, for example, to verify security vulnerabilities or they can be informed about possible problems such as restricted or interrupted accessibility of the pivot system or other systems, and rectify these problems immediately.

The second aspect – constant, uninterrupted availability of the pivot system – in particular is vitally important in this module since all tests are carried out via this system.

Tip by Sebastian Schreiber

If you set up "artificial access" on one of your DMZ systems for this test module, always take care to ensure that this access can only be used from SySS IP addresses, e.g. by means of an access control list!

2.14.5 TERMSERV: Security of remote access solutions

Summary

Within the framework of this module, remote access solutions are examined for possible vulnerabilities. Both authentication and the privilege concept are the focal points in this respect. For example, it can be tested during a "breakout analysis" whether access to applications other than those intended is possible or whether other resources in the company network can be attacked from the terminal server through different privilege escalations.

Figure 2.43: Module TERMSERV

Starting situation

A large number of companies offer their employees or external service providers restricted remote access to selected internal resources. Current technical solutions here include Citrix XenApp, Microsoft Remote Desktop Gateway or VMware Horizon. Since these technologies therefore represent an interface which is accessible from the internet in a company's own company network, great importance should be attached to their security level. Based on different test perspectives, SySS evaluates in this module whether there are any vulnerabilities in the implementation of the remote access solution.

Objective

The objective of this test module is to identify ways in which an attacker can use a remote access solution to access critical company resources without authorization. An attempt is made here, for example, to bypass authentication, cancel defined user guidelines, break out from the context of individual applications or attain other privilege escalation. If requested, SySS will also attempt to identify and then compromise other network resources which are accessible from the terminal server. The objectives of the test module are to identify potential and specific vulnerabilities and to recommend measures whose implementation leads to optimum hardening of the remote access solution. In some cases, however, it will only be determined whether an individually provided application allows access to the underlying system. It is important here to specifically define the question that is relevant to you.

Implementation

Depending on the technology to be tested and the adopted test perspective, SySS will consider the following aspects for example:

– Attacks at system level (see IP-RANGE module in Section 2.1 on Page 23)
– Attacks against authentication (1-factor, 2-factor, etc.)
– Breakout from individual applications
– Privilege escalation on the terminal server or within the virtual desktop environment
– Network-based attacks against other systems

For this purpose, SySS uses, for example, breakout techniques via system dialogs, unlocked keyboard shortcuts or on-board resources and, if possible, self-written attack scripts in order to ideally gain access to a command line such as CMD or the PowerShell. If this is successful, other test activities such as password-guessing attacks against other user accounts are favored.

If administrator privileges or system privileges are also acquired on the local system using local privilege escalation techniques, other interesting information can be extracted from the memory, file system or registry. Based on this privilege status, network-based attacks can also be carried out more efficiently.

SySS also checks whether it is possible to develop an alternative communication channel, e.g. a reverse shell to a SySS root server on the internet, or whether there are any other ways to exfiltrate internal company data via the remote access solution (e.g. copy and paste via the clipboard, etc.).

Cooperation by the customer

Test preparation: Before the start of the test, in the KICKOFF, SySS will discuss with the customer the required attack scenarios and the test scope, and request access data for the remote access solution.

Contact person: With this test module, it is also important that the contact person can be contacted by phone to ensure, for example, that blocked access can be unblocked again immediately. Some remote access solutions allow access to a terminal server on which several users are accommodated simultaneously. If, contrary to expectations, availability restrictions occur due to the test activities, the contact person will be informed immediately and can carry out corrective measures.

Tips by Sebastian Schreiber

Do not underestimate the time required to make preparations for this test module! Depending on the size of the company, it may well take several days to apply for remote access. A second authentication factor (e.g. a hardware token) must also often be provided by mail.

2.14.6 REVIEW: Security evaluation of concepts, processes, documents and organizational requirements

Summary

SySS evaluates existing security concepts and architectures in order to draw attention early on in a project phase to hitherto disregarded risks. Security-related internal processes and documentation, as well as organizational requirements can also be critically examined during this test module.

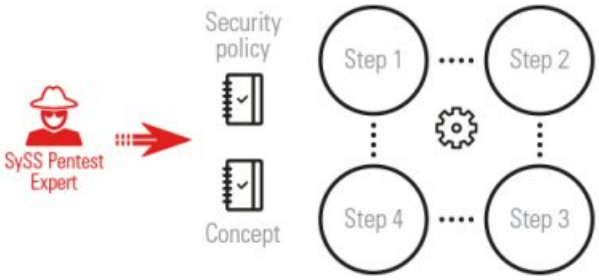

Figure 2.44: Module REVIEW

Starting situation

IT security can only be guaranteed when it is regarded as a process[4], but not through purely selective measures. SySS therefore also offers tests of the organizational requirements which define IT security. These requirements may include, for example, security guidelines and security manuals, but also sets of rules within the IT infrastructure. The material to be tested is provided to our consultants who then familiarize themselves with it and recommend improvements. On request and where practical, the reviews can be accompanied or completed by meetings or workshops. This module does not cover the technical inspections during security tests. In order to control the status of the actual implementation of specifications or security guidelines, we recommend that each suitable module from this white paper be examined.

This test module is also practical as part of development projects involving new applications (web applications and services, mobile apps, etc.) in order to evaluate the formulated security concepts and architectures. Consequently, attention can be drawn at an early stage to any security risks which were not yet considered.

Objective

The objective of the project varies depending on the item to be evaluated. In principle, however, SySS will attempt to show potential improvements for increasing the target security level. Just like in technical security analyses, the viewpoints and mindsets of an attacker are adopted here in order to identify possible vulnerabilities.

[4] Schneier, May 2000, `https://www.schneier.com/crypto-gram-0005.html`

Implementation

Implementation of a review largely depends on the nature of the project. If it is necessary, for example, to evaluate security guidelines, it is often sufficient to carry out a thorough examination and revision of documents. However, an on-site workshop is ideally suited for evaluations of architectures or finished concepts as it initially provides the participants – primarily software architects and developers – with an opportunity to present the subject of the evaluation and identify any potential risks afterwards during a joint discussion. Evaluations of security-related processes and procedures in turn are ideally carried out in the form of interviews with each responsible person.

Cooperation by the customer

Test preparation: Depending on the form of the review, corresponding responsibilities must be clarified in advance in order to appoint ideal interview partners or workshop participants. Planning of a workshop should start at an early stage since it is often necessary to find a date that works for many people. If documents are to be examined, the respective documents must be sent to SySS in the version to be evaluated in good time before the start of the project.

Contact person: In an ideal scenario, a central contact person should be appointed to plan the review. This person will be available to answer questions, gather any necessary information or coordinate meetings.

Tip by Sebastian Schreiber

Combine the early, conceptional security analysis with a technical security analysis which is directly connected to implementation of the project. You can therefore cover both important security risks in the concept and classic vulnerabilities during its implementation.

2.14.7 Special, individual test focal point

If your concern is not covered by the test modules presented in this white paper, do not hesitate to call us to explain it in detail. Thanks to our pentest architects, we will find a solution for you in nearly every case, based on our many years of experience and expertise in practically every consulting area of IT security. On request, we will also be pleased to stage a joint workshop with you in order to design a possible test project.

3 Red Teaming

Red teaming is a test and training method which originally comes from the armed forces and was first used in the corporate world in the early 2010s. In this method, a team of attackers ("red team") is specifically deployed to test the defense level of the company and to train the corresponding team of employees ("blue team").

The employees only know that they may be attacked at any time. This increases awareness significantly and the employees detect more actual attacks due to the permanent search for the attackers' team (red team).

Figure 3.1: RED TEAMING

3.1 Red teaming procedure

The objective of a traditional penetration test is to find all vulnerabilities in a clearly defined narrow framework which shows the test object, to document these vulnerabilities and to formulate recommendations on how to fix them.

However, the objective of a red teaming assessment is to carry out a wide-ranging attack with the aid of freely selectable attack vectors. The framework here is very broad and the attacker has a great deal of creative freedom. The attacker not only uses the technical methods which are available to him/her, but also sometimes sounds out whether he/she can gain access to sensitive data through social engineering and find ways to penetrate the company. The task of a deployed blue team is to detect and prevent attacks by the red team. The blue team and its defensive mechanisms are tested and further enhanced in this case. This test method is also used to test whether an attacker can succeed in a specific time period to compromise a company network from an external perspective.

Starting situation

In the majority of critical areas in the corporate world, there are exercises to rehearse an emergency. In the area of IT security, these exercises have only been available sporadically to date and not as an integrated scenario.

Red teaming enables a real attack to be simulated completely. It tests whether the employees have adequate training both in terms of technology and awareness, and whether the defined emergency processes work properly.

In this simulated approach, employees in internal IT security receive training in how to detect targeted advanced persistent threat (APT) attacks. A red teaming assessment may also be practical in companies that do not have any dedicated IT security personnel since the technical defense mechanisms can be evaluated in regard to their effectiveness during such a test.

Red teaming is stipulated as a test method for banks according to the TIBER-EU framework published in May 2018, respectively the TIBER-DE framework from July 2020. Red teaming tests must therefore be used in this sector.

Objective

The main focal points of red teaming is to educate the blue team and provide the team members with advanced training using a games-based approach. However, this type of test may also be used for other objectives, as shown in the following examples:

– Review of the company's overall security based on a wide-ranging attack portfolio
– Simulation of a real attack in order to verify the effectiveness of the current technical protective measures, to test employees and their actions, and to evaluate which processes exist and whether they are observed
– Training of the blue team

Implementation

Implementation of a red teaming assessment is individually adapted to existing customer needs. Roughly speaking, the test contains the following eleven phases. Focal points can be defined in the initial workshop.

Kick-off workshop

The extent of the red teaming project is evaluated together with the customer during a workshop. Based on the following aspects, an initial impression of the project schedule can be gained. SySS will examine any individual requests and questions here.

Digital public footprint

During an initial phase, all kinds of publicly accessible data are analyzed to identify possible targets for subsequent attacks from the internet and with social engineering methods.

Information gathering

The objective of this phase is to obtain the most realistic picture of the visible attack surfaces of systems and services which are accessible from the internet. For this purpose, an attempt is made to obtain as many details as possible concerning the utilized service and operating system versions.

Persistence in the company network

After SySS has gained temporary access to the company's internal network, an attempt is made to create a permanent connection. More care is taken here than in the previous phases to ensure that the least invasive approach is used in order to remain undetected by detection and defense systems.

Social engineering

If it is impossible to gain access to the internal company network through the above-mentioned technical phases, social engineering methods are used to acquire access data. Due to the ethical challenges of social engineering and the simultaneously high success rate, it is advisable to appoint an employee who simulates the behavior of a victim and clicks, for example, on the link in a phishing e-mail.

Compromising of systems and services

Irrespective of whether compromising initially took place via server systems or client systems, an attempt should be made to compromise other systems in the internal network so that the privileges in the internal network can be extended using the acquired data and findings as the test proceeds.

Privilege escalation

The data and information compiled in the preceding phases typically enable an attacker to move around in the network using the restricted privileges of a standard user. An attacker may also have extended privileges regarding individual systems or services. During this phase, an initial attempt is made to extend the local privilege status. This is then followed by an analysis of what other attack possibilities arise through the use of a standard user account in the network.

Achieving defined objectives

This phase describes target attainment of the project objective agreed in the workshop. If data is involved here, it is identified and extracted using the most minimally invasive methods. It also analyzes whether monitoring detects anomalies in network traffic.

Triggering of protective systems and processes

If the agreed project objective was attained and the attack has not yet been noticed, other methods with which the objective can also be achieved are determined. The aggression of the attacks is continuously increased in this respect in order to determine the level at which internal protective systems and processes are effective. In particular, it is tested here whether the attack can be repelled successfully and quickly.

Rectification of advanced persistent threat simulation

During this phase, the effectiveness of the defensive measures and the emergency concept will be tested. By gaining access during the preceding phases, the attacker has managed to acquire various privileges for different systems. The customer's objective is to remove the attacker as completely as possible from the company network and close all backdoors which were already open. In addition to the measures of the blue team, the attack is continued by the red team. To this end, use is made of techniques which are hard to detect and enable continuous and secure access to the network.

Documentation

In this phase, the results of the test are summarized chronologically in written documentation. The documentation corresponds to the SySS standard and is quality-assured in two stages. The results are also presented and explained in a final presentation adapted to the particular target group.

Cooperation by the customer

Red teaming requires intensive support from the customer. Our many years of experience in performing red teaming assessments have shown that status phone calls at least every two weeks contribute significantly to the success of the test. SySS should also be included in the test preparations.

Common goals should be defined here and critical systems should be identified. Due to the fact that red teaming is a complete black box test, SySS can only determine after an initial scan what types of systems are contained in a network segment. However, it is possible that this scan may lead to malfunctions in outdated systems. It is therefore recommended that a whitelisting approach be used with critical infrastructures. In this case, SySS receives a list in advance containing subnetworks which are not critical.

If SySS has access to other network areas during an assessment, further action must be agreed in detail. During implementation, it is discussed in detail what information will be passed on to the blue team. In individual test phases, passing on information should be avoided entirely so that it can be verified whether or not attacks are detected. In the event of very advanced compromise, it may be practical to purposefully pass on information during the "Rectification of advanced persistent threat simulation phase".

3.2 Purple teaming

As previously outlined, red teaming tests are highly suitable for verifying how well prepared a company's IT security is for also detecting and preventing attacks.

If the blue team is still being set up, it is recommended that the "purple teaming" approach be used. In this approach, the blue team and red team directly interact with each other. In addition, the blue team initially receives support. If this support is not required, steps 1-3 of the procedure described below may be omitted.

In order to optimally implement a purple team assessment, SySS recommends the following procedure:

Step 1

Before a red teaming test is performed, the SySS employees working in digital forensics and incident response will discuss their incident detection methods with the customer. This means that improvements can already be devised in a workshop.

Step 2

For the second step, SySS recommends testing the implemented changes and/or the current situation. For example, this can take the form of a role play. Example scenarios are used here to run through processes and this results in the emergence of important insights. The scenarios are based on the results of the initial workshop. Alternatively, the red team provides a description of various scenarios from which a selection can be made.

Step 3

Once the processes are more sophisticated, testing them in practice in a productive environment is recommended. For this purpose, the customer's blue team will continue to receive support from the employees of SySS working in digital forensics and incident response, while the red team carries out the scenarios discussed beforehand. There are detailed discussions about what the blue team has detected and which attack vectors have been carried out by the red team, depending on each scenario. Consequently, the blue team is constantly developing improved routines to quickly detect attacks and to defend against them.

Step 4

The amount of support provided to the customer's blue team is reduced and the attack scenarios by the red team become increasingly complex. This allows the blue team to be autonomous. Following this step, it is recommended to carry out a separate red team assessment in order to fully test the implementation of the incident detection and incident response measures.

Network monitoring and other types of monitoring during a purple teaming assessment produce data which can be subsequently evaluated and used to optimize the customer's detection methods (SIEM, log evaluations, etc.). The services of the SySS red team and DFIR department complement each other here so that the customer can identify any found vulnerabilities, react as quickly as possible and prepare their own defense against these vulnerabilities. In an ideal scenario, purple team assessments are performed frequently and used for different scenarios so that the customer's IT security can be continuously developed and there is a routine response in the event of a real incident. This also improves security in the long term and shortens the duration of an incident.

3.3 Ethical principles for social engineering

SySS implements social engineering (SE) projects. These projects are only implemented by specially qualified and aware employees who have also received training beforehand regarding legal and ethical aspects.

We realize that an SE test specifically exploits people's weaknesses. An SE test is only carried out if a) there is no other possibility to perform the test in another way, and b) SE appears suitable as a method. SySS employees act very responsibly, cautiously and circumspectly here. Social engineering techniques are used to check awareness measures or attain the objective of a red teaming assessment. In social engineering projects, our consultants observe the ten rules that we have formulated:

1. The privacy of the customer's employee is protected.
2. Tests must be announced or already form part of corporate culture.
3. SySS does not mention any names of employees who stood out during the test.
4. Only techniques that the corresponding employees can expect in their normal work are used.
5. Communication with employees is minimized. Passive action is always preferred.
6. Every action stems from the two ethical perspectives of "utilitarianism" and "deontology".[1] After weighing up these perspectives, an action is selected or rejected.

[1] Utilitarianism (benefit/advantage): An action is certainly morally correct whenever the overall benefit, i.e. the general welfare of everyone affected is maximized. Deontology: Protects the individual more than it protects the community. Is the action right or wrong for the individual? For example, the following comment could be made to an employee of the company to be tested during a SE test: "Your child has had a serious accident". Utilitarianism regards the content of this comment as follows: If this resulted in a security measure being circumvented (e.g. a guard leaves their post), this information helps the company -> measures must be improved; according to deontology, such action would be considered absolutely morally wrong.

7. All utilized social engineering techniques are non-violent.
8. Only devices intended for purely commercial purposes may be removed.
9. The tests are carried out for the benefit of the company. The objective of the tests is to uncover vulnerabilities in the processes and/or the inadequate effectiveness of awareness measures. It is never a question of proving inadequacies of an individual employee.
10. The action is not destructive. Lock picking, for example, is only used in exceptional cases.

The following techniques may be used, for example, during social engineering assessments:

– Phishing and spear phishing e-mails
– Pre-texting
– Calls and text messages
– Tailgating
– Copying employee IDs
– Using a disguise and presenting a false identity
– Sending letters
– Copying signatures, having obtained written permission from the person concerned
– Searching for persons using social community profiles
– Stealing unattended security tokens/computer systems

The techniques to be used during a project are agreed beforehand in a workshop. The particular advantages and disadvantages are also pointed out during this workshop.

4 About SySS

4.1 Company history

SySS GmbH was founded in 1998 by Sebastian Schreiber, who holds a degree in computer science, in order to offer high-quality security tests.

SySS has six business areas. In addition to penetration and red teaming tests, we also provide digital forensics/incident response, technical consulting, live hacking and training.

SySS GmbH's head office is based in Tübingen in south-western Germany, and the company has offices in Frankfurt/Main and Munich. SySS GmbH also has an Austrian subsidiary, SySS Cyber Security GmbH. The customers of SySS are companies of all sizes and from all industries. They include both a large number of medium-sized enterprises and German DAX groups.

SySS gives technical presentations at national and international congresses in Germany and European and non-European countries.

Employees of SySS frequently feature as experts in various print, radio and online media, e.g. Der Spiegel, Die Zeit, Financial Times Deutschland, Stuttgarter Zeitung, Süddeutsche Zeitung, ARD, ZDF, Südwestrundfunk, Hessischer Rundfunk, RTL, Pro7 and CHIP TV.

4.2 Fundamental ethics for penetration testers

Based on already existing codes and empirical values collected over many years, SySS instigated the first initiative to formulate fundamental ethical principles for penetration testers. These ethical principles were published for the first time in issue 04/2009 of the IT journal "Datenschutz und Datensicherheit" (DuD) and reflect the attitude and foundations of work at SySS. We organize our work based on the following ethical principles:

- **Independence:** Companies contracted to perform penetration tests only do so in firms where they were not involved in the design of the IT infrastructure or the implementation of security measures and to whom they did not sell or want to sell their own software. This is the only way of ensuring that the test results are objective.
- **Confidentiality:** Both the identity of the commissioning company and all insights into internal networks, structures and all data – even if they are made available to the penetration tester – must be treated in absolute confidence.
- **Prohibition of commissions:** It is prohibited to accept commissions or comparable benefits.
- **Caution:** The customer must be informed about potential risks which may occur during the tests.
- **Professionalism and quality management:** Work must be carried out professionally and must be subject to quality management. The penetration tester performs their work to the best of their professional knowledge and according to an ethical conscience.
- **Liability:** Contractually assured promises and verbal promises made during consultations must be kept with binding effect by the employees of the company performing the penetration tests.
- **Objectivity, neutrality and transparency:** Conclusions must be objective and presented in a comprehensible way.

- **Conflicts of interest:** Conflicts of interest between penetration testers and customers must be avoided and indicated and eliminated if necessary.
- **Strict legality principle:** The laws of the countries in which the penetration tests are performed must be strictly observed even if partial results of a penetration test might represent a conflict of interest with the existing legislation. For example, the detection of vulnerabilities may favor infringements of existing law in certain cases. Penetration testers are therefore obliged to become familiar with the particular legal situation and take special care to ensure that their work is performed with the limits laid down by the law.
- **Respect for people:** Social engineering projects and attacks against human behavior – if they are actually realized – will only be carried out after prior notification.
- **Correct quotation:** If external expertise is used during the work, the sources or copyright holders must be shown correctly.

5 Selected SySS Publications (since 2012)

SySS regularly publishes articles in journals, on online platforms or as part of congresses. A selection of these articles can be found below. For further information about our publications, visit: https://www.syss.de/pentest-blog/category/know-how/ and https://www.syss.de/pentest-blog/pentest-library/.

Staller, Nicola: 6 Maßnahmen gegen Passwortrateangriffe (6 measures against password-guessing attacks). In: Protector 9/2024: https://www.syss.de/fileadmin/dokumente/Publikationen/2024/2024_09_30_Staller_Sonderdruck_Massnahmen_gegen_Passwortrateangriffe.pdf

Abrell, Moritz: SIP Digest Leak: Angriff auf SIP-Konten (Attacks on SIP accounts). In: VAF REPORT 01/2022: https://www.syss.de/fileadmin/dokumente/Publikationen/2022/2022-05-12_Abrell_Sonderdruck_VoIP-Hack_VAF-Report.pdf

Bechler, Moritz: Oracle Native Network Encryption – Breaking a Proprietary Security Protocol. SySS publication, December 2021: https://www.syss.de/fileadmin/dokumente/Publikationen/2021/2021_Oracle_NNE.pdf

Deeg, Matthias/Klostermeier, Gerhard: On the Security of RFID-based TOTP Hardware Tokens – Hacking NFC-enabled Time-Based One-Time Password Hardware Tokens. SySS publication, June 2021: https://www.syss.de/fileadmin/dokumente/Publikationen/2021/2021_06_21_Deeg-Klostermeier_On_the_Security_of_TOTP_Hardware_Tokens.pdf

Bostanov, Vladimir: Client Puzzle Protocols as Countermeasure against Automated Threats to Web Applications. In: IEEE Access, May 2021: https://www.syss.de/fileadmin/dokumente/Publikationen/2021/2021_05_19_Bostanov_CPP4WebApp.pdf

Ritter, Christoph: Angriff auf Anti-Phishing-Banner in E-Mails – Eine Warnung, die alles schlimmer macht (Attack on anti-phishing banners in e-mails – A warning that makes everything worse). SySS publication, April 2021: https://www.syss.de/fileadmin/user_upload/2021_04_Angriff_auf_Anti-Phishing-Banner_in_E-Mails.pdf

Abrell, Moritz: New Ways Of Communicating – When End-To-End Encryption Gains a New Meaning. SySS publication, June 2020: https://www.syss.de/fileadmin/dokumente/Publikationen/2020/2020_07_28_New_Ways_of_Communicating_When_End-to-End-Encryption_Gains_a_New_Meaning.pdf

Krauß, Thomas: Herausforderungen für die IT-Sicherheit bei der Elektromobilität und autonomem Fahren (Challenges to IT security in electromobility and autonomous driving). In: Informatik Aktuell 8/2019: https://www.informatik-aktuell.de/betrieb/sicherheit/herausforderungen-fuer-die-it-sicherheit-bei-der-elektromobilitaet-und-autonomem-fahren.html

Lutz, Torsten: Mehr Sicherheit in SAP Town (More security in SAP town). In: Protector & WiK 6/2019: https://www.syss.de/fileadmin/dokumente/Publikationen/2019/2019_06_14_SAP_Lutz.pdf

Bechler, Moritz: LDAP Swiss Army Knife – A directory server for LDAP client analysis and exploitation. SySS publication, May 2019: https://www.syss.de/fileadmin/user_upload/2019_05_LDAP_Swiss_Army_Knife.pdf

Schreiber, Sebastian: Internet der Dinge – Smart genug? (Internet of Things – smart enough?) In: Protector & WiK 3/2019: https://www.syss.de/fileadmin/dokumente/Publikationen/2019/2019_01_22_IoT_Schreiber.pdf

Buchegger, Philipp: Hacking Fingerprint Readers Without Making a Mess – Using tin foil instead of human skin. SySS publication, November 2018: https://www.syss.de/fileadmin/dokumente/Publikationen/2018/Hacking_Fingerprint_Readers_without_Making_a_Mess.pdf

Vollmer, Dr. Adrian: Antivirus Evasion With Metasploit's Web Delivery – Leveraging PowerShell to Execute Arbitrary Shellcode. SySS publication, July 2018: https://www.syss.de/fileadmin/dokumente/Publikationen/2018/Antivirus_Evasion_Metasploit.pdf

Deeg, Matthias/Klostermeier, Gerhard: Rikki Don't Lose that Bluetooth Device – Exploiting the Obvious: Bluetooth Trust Relationships. SySS publication, July 2018: https://www.syss.de/fileadmin/dokumente/Publikationen/2018/Rikki_Dont_Lose_That_Bluetooth_Device.pdf

Deeg, Matthias/Klostermeier, Gerhard: Case Study: Security of Modern Bluetooth Keyboards – SySS IT Security Research Project. SySS publication, June 2018: https://www.syss.de/fileadmin/dokumente/Publikationen/2018/Security_of_Modern_Bluetooth_Keyboards.pdf

Vollmer, Dr. Adrian: Angriffe auf RDP – Wie man RDP-Sitzungen abhört (Attacks on RDP – How to eavesdrop RDP sessions). SySS publication, November 2017: https://www.syss.de/fileadmin/dokumente/Publikationen/2017/2017_11_07_Vollmer_Angriffe_auf_RDP.pdf

Schreiber, Sebastian/Straßheim, Alexander: IoT-Penetrationstest (IoT penetration test). In: DuD Datenschutz& Datensicherheit 10/2017 : https://www.syss.de/fileadmin/dokumente/Publikationen/2017/2017_09_07_Strassheim_Schreiber_IoT-Penetrationstest__DuD.pdf

Schreiber, Sebastian: Penetrationstests in der IT – Angreifbare Schwachstellen finden und schließen. (Penetration tests in IT – finding and closing attackable vulnerabilities) In: unternehmermagazin 3/4, 2017: https://www.syss.de/fileadmin/dokumente/Publikationen/2017/2017_09_14_UMAG-03-04-2017-TT-24-25-Schreiber.pdf

Scholl, Edgar/Schreiber, Sebastian: Leider gehackt. (Unfortunately hacked). In: impulse 07+08, 2017: https://www.syss.de/fileadmin/dokumente/Publikationen/2017/2017_07_01_Leider_gehackt.pdf

Deeg, Matthias/Klostermeier, Gerhard: Of Mice and Keyboards – On the Security of Modern Wireless Desktop Sets. SySS publication, June 2017: https://www.syss.de/fileadmin/dokumente/Publikationen/2017/2017_06_01_of-mice-and-keyboards_paper.pdf

Nerz, Sebastian: Alltag und Arbeitsfelder der IT-Forensik (Everyday work and work areas in IT Forensics). In: IT-Sicherheit 2/2017: https://www.syss.de/fileadmin/dokumente/Publikationen/2017/2017_05_Alltag_und_Arbeitsfelder_der_IT-Forensik_IT-SICHERHEIT_2_2017.pdf

Vollmer, Dr. Adrian: Attacking RDP – How to Eavesdrop on Poorly Secured RDP Connections. SySS publication, March 2017: https://www.syss.de/fileadmin/dokumente/Publikationen/2017/2017_03_13_Attacking_RDP.pdf

Grasmück, Dr. Oliver/Mangold, Marcel: Safety first! In: smart engineering 1/2017: https://www.syss.de/fileadmin/dokumente/Publikationen/2017/2017_02_17_Safety_First_smart_engineering_1_17.pdf

Schreiber, Sebastian: Schwachstellen vor dem Hacker finden (Finding vulnerabilities before the hacker). In: Energie & Management 7/2016: https://www.syss.de/fileadmin/dokumente/Publikationen/2016/2016_07_04_Schwachstellen_vor_dem_Hacker_finden.pdf

Schreiber, Sebastian: Penetrationstests für Stadtwerke – Den Hacker nicht ins Netz lassen (Penetration tests for municipal utilities – don't let the hacker into the network) In: ew Spezial 2/2016: https://www.syss.de/

fileadmin/dokumente/Publikationen/2016/2016_05_23_Hacker_nicht_ins_Netz_lassen_ew_Spezial_02-2016.pdf

Stühler, Roman: Schadcode auf Smartphones – wie sicher sind Android-Geräte vor Angriffen? (Malicious code on smartphones – how secure are Android devices against attacks)? SySS publication, March 2016: https://www.syss.de/fileadmin/dokumente/Publikationen/2016/2016-03_03_Schadcode_auf_Smartphones.pdf

Deeg, Matthias: Verantwortungsvoller Umgang mit Sicherheitsschwachstellen (Responsible handling of security vulnerabilities). SySS publication, December 2015: https://www.syss.de/fileadmin/dokumente/Publikationen/2015/2015_12_02_Verantwortungsvoller_Umgang.pdf

Deeg, Matthias: Deactivating Endpoint Protection Software in an Unauthorized Manner. Conference Paper, DeepSec, Vienna, November 19, 2015: https://www.syss.de/fileadmin/dokumente/Publikationen/2015/Deactivating_Endpoint_Protection_Software_in_an_Unauthorized_Manner_-_DeepSec_2015.pdf

Deeg, Matthias: Privilege Escalation via Client Management Software. Conference Paper, BSidesVienna 0x7DF, Vienna, November 21, 2015: https://www.syss.de/fileadmin/dokumente/Publikationen/2015/Privilege_Escalation_via_Client_Management_Software_-_BSidesVienna_2015.pdf

Borrmann, Micha: Attacking all your IPv4 devices at home from the Internet via Dual-Stack Lite. Hacktivity, Budapest, October 10, 2015: https://hacktivity.com/en/downloads/archives/397/

Steglich, Finn/Straßheim, Alexander: Digitaler Kassenraub (Digital cash till robbery). Austricksen von In-App-Bezahlfunktionen (Outsmarting in-app payment functions). In: iX 7/2015, Pp. 52–55: http://www.heise.de/ix/inhalt/2015/7/52/

Nerz, Sebastian: IT-Sicherheit und die EU-Datenschutznovelle: Worauf deutsche Unternehmen sich einstellen müssen (IT security and amendment of EU data protection: What German companies must face up to). SySS publication, May 2015: https://www.syss.de/fileadmin/dokumente/Publikationen/2015/IT-Sicherheit_und_die_EU-Datenschutznovelle.pdf

Deeg, Matthias: Privilege Escalation via Client Management Software. SySS publication, April 2015: https://www.syss.de/fileadmin/dokumente/Publikationen/2015/Privilege_Escalation_via_Client_Management_Software.pdf

Deeg, Matthias/Nerz, Sebastian/Sauder, Daniel: Outsmarted – Why Malware Works in face of Antivirus Software. SySS publication, August 2014: https://www.syss.de/fileadmin/dokumente/Publikationen/2014/Antivirus_Evasion_engl.pdf

Borrmann, Micha: Thunderbird gibt falschem Absender das Echtheits-Siegel (Thunderbird gives the wrong sender the seal of authenticity). In: c't 17/2013, S. 16: http://www.heise.de/security/meldung/Thunderbird-gibt-falschem-Absender-das-Echtheits-Siegel-2044405.html

Borrmann, Micha: Microsofts Hintertür – Zweifelhafte Updates gefährden SSL-Verschlüsselung (Microsoft's back door – dubious updates endangering SSL encryption). In: c't 17/2013, S. 16: http://www.heise.de/ct/ausgabe/2013-17-Zweifelhafte-Updates-gefaehrden-SSL-Verschluesselung-2317589.html

Schreiber, Sebastian: Komplexität bildet das Hauptproblem. (Complexity is the main problem). In: isreport 10/2012

Schreiber, Sebastian: Wir bemerken eine zunehmende Professionalisierung der Angreifer (We are seeing increasing professionalization of attackers). In: Bankmagazin 10/2012

Schreiber, Sebastian: Windows 8 – Der richtige Weg (Windows 8 – the right path). In: CHIP 8/2012

Heitmann, Kirsten/Schreiber, Sebastian: Sicherheit bei Web-Shops (Security in Online Stores). In: Ecommerce Vision 5/2012: http://www.ecommerce-vision.de, May 16, 2012

THE PENTEST EXPERTS

SySS GmbH
Tübingen • Germany
+49 (0)7071 - 40 78 56-0
info@syss.de

SySS Cyber Security GmbH
Vienna • Austria
+43 (0)50 - 7977-0
info@syss.at

WWW.SYSS.DE
WWW.SYSS.AT