senseposts

Security Assessment Methodologies

1. Introduction

SensePost is an information security consultancy that provides security assessments, consulting, training and managed vulnerability scanning services to medium and large enterprises across the world. Through our labs we provide research and tools on emerging threats. As a result, strict methodologies exist to ensure that we remain at our peak and our reputation is protected.

An information security assessment, as performed by anyone in our assessment team, is the process of determining how effective a company's security posture is. This takes the form of a number of assessments and reviews, namely:

- Extended Internet Footprint (ERP) Assessment
- Infrastructure Assessment
- Application Assessment
- Source Code Review
- Wi-Fi Assessment
- SCADA Assessment

2. Security Testing Methodologies

A number of security testing methodologies exist. These methodologies ensure that we are following a strict approach when testing. It prevents common vulnerabilities, or steps, from being overlooked and gives clients the confidence that we look at all aspects of their application/network during the assessment phase. Whilst we understand that new techniques do appear, and some approaches might be different amongst testers, they should form the basis of all assessments.

2.1 Extended Internet Footprint (ERP) Assessment

The primary footprinting exercise is only as good as the data that is fed into it. For this purpose we have designed a comprehensive and exhaustive methodology that leverages some of the popular and stable Internet search engines in the form of data scraping.

Search Engine Crawling

Extracting host, domain and auto Whois registrant information:

- Google (Scraping)
- Yahoo (BOSS API)
- Bing API
- CUIL (Scraping)

Image Searches

Image searches are becoming more mature and several search engines offer a service whereby an image can be uploaded and compared to similar images already indexed on websites. This is very useful to identify sites where either copyrighted material is being hosted or where a site has been recreated for phishing purposes. Image search services include:

- Google Image search utilizing Goggle's
- Specialized Image search engines such as TinEYE

Image searches also allow for more directly identifying sites where a domain may be mentioned. However, it is not as straightforward as merely copying an image, loading it into the search engine and then waiting for results. Knowledge about image technologies and a simple understanding about the difference between a GIF or JPG can mean the difference in finding 10 or 2 websites.

DNS Repository Searches

There are several DNS or related repositories on the Internet where more elaborate DNS related data can be obtained. Data found in these repositories include reverse and forward DNS lookup data, IP ownership (netblocks), shared nameservers and virtual hosts.

- Robtex and Alexa – web information sites with special Internet traffic analysis
- Netcraft – probably the foremost repository of arbitrary DNS and website data
- HackRack (for those clients who make use of the HackRack service)
- Binger Sweeps – utilizing Bing IP searches to scan country IP ranges

SensePost has special relationships with many of these data repositories, which puts us in a unique position to have access to web data normally not reserved for any Internet user.

Client Provided Information

An Extended Internet Footprint (ERP) is not meant to be a blackbox or unassisted exercise. A more comprehensive and realistic Footprint Assessment will be possible when a client can provide additional data regarding its domains, subsidiaries, hosting service providers and trusted third parties. Prior to commencing the assessment, a detailed list of data requirements will be forwarded to the client.

Data that the client can contribute to a Footprint Assessment includes:

I) Known network information

- Domains used by the client and associated third parties and subsidiaries

- Hosted environments used by the client and third parties authorised to market and distribute associated client data.

ii) DNS registrant / Registrar information

- Known registrant names used to register domain names
- Administrative contact information
- List of known thirrd party service providers
- Images and branding

iii) Internal log information (only referring to domains)

- Web server logs (referrer checking)
- Proxy logs
- Mail logs (domains set to and received from)
- SSL certificate information (where possible)

Footprinting (Scanning)

SensePost has developed a technology called Yeti, where an assortment of DNS related information can be added that will automate the majority of the steps that are required to complete a picture of a client's Internet footprint.

Obtain Domains

Objective:

Domains belonging to the organization are collected and collated during the preparatory phase and fed into Yeti. Other methods used include:

- Inspection of links to and from the client website
- Top level domain expansion
- Who is service wildcard expansion

Output:

A list of domains related in some way or another to the organization. Domains may be deemed related in various ways, for example, by using the same branding, registered by the same person, similar domains in other countries or more.

Determine IP Addresses

Objective:

- Obtain a list of possible IP addresses that are used by the organization

From the previous phase, all domains found will be used to determine possible IP addresses. Methods used include:

- Zone transfers
- Brute force DNS lookups
- Reverse DNS scans

Output:

A list of IP addresses and related DNS names for each domain.

Determine IP Network Blocks

Objective:

- Determine boundaries (and supporting information) on the IP address ranges

From the previous phase, the list of IP numbers and domains are used to determine:

- Routing blocks
- Start/stop network block information
- Geo-location

Output:

The list of IP addresses is expanded to ranges. The ranges are inspected to determine the routing blocks as well as geo-locations.

Active IP Network Block Determination

Objective:

- Reduce currently identified network blocks to blocks that are actively being used.

IP ranges obtained form the previous phase are used as input to determine active IP network blocks. Tools used include:

- ICMP broadcasts

- Traceroute separation

Output:

A subset of the previously identified network ranges, which represents blocks that are actively in use.

Vitality

Objective:

- Determine which hosts on the specified subnets are “alive”

Confirmed IP ranges are probed to determine hosts that are visible or “alive”. The following tools and techniques are used to this extent:

- ICMP ping sweep
- TCP ping
- Mini port scan

Output:

List of IP addresses that are visible and reachable from the Internet.

Consolidation

Objective:

- Consolidate and collate information discovered in previous steps
- Ensure that the assessment information is meaningful and relevant

Output:

Consolidated view of the information gathered in the assessment. The information gathered during the preparatory phase and subsequently fed into the automated scanning phase should yield results which is more substantive than a simple Whois / DNS / Google inurl: search.

Interactive Web Applications

Objective:

- Identify all IPs where interactive web applications are active
- Identify the components and framework of the web applications

Output:

List of IPs, hostnames, URLs, web server type, and the application framework used in the make up of the web application.

This information can be used to determine whether the applications are owned or managed by a third party, comply with branding requirements, and perhaps pose a risk to the corporation as a result of its underlying technology and architecture.

2.2 Infrastructure Assessments

SensePost follows a strict methodology to ensure that a structured process is followed when conducting an Infrastructure Security Assessment. It provides the client with a baseline against which the quality of the assessment can be measured. The specific aspects that are assessed include:

System Enumeration and Information Gathering

- Perform a full network survey to determine attack surface area. This includes harvesting:
    - Domain names
    - Server names
    - IP addresses
    - Network maps
    - OS Identification
    - Network device identification
- Full network enumeration using scanning techniques
    - List of all open, closed and filtered ports
    - IP addresses of live systems
    - List of discovered protocols
    - Determine potential threats and risks
    - Understand system design and operation
- Perform a vulnerability analysis assessment against all identified hosts
    - For whitebox testing exclude scanning system from IPS/IDS technologies

- For blackbox testing perform IPS/IDS evasion techniques
- Perform an exhaustive system service identification
- Perform vulnerability scanning to determine flaws within various OS platforms and OSI layered technologies
- Testing for known issues regarding versions implemented throughout infrastructure
- Verify all reported patch levels
- Testing default software configuration flaws
- Testing for weak and default credentials for various technologies
- Verify scanning results though manual testing, service detections and version enumeration verification
- Perform false positive detection against results from vulnerability assessment phase

- Exploitation of issues after vulnerability verification
  - If in scope, exploit known weaknesses
  - Gaining access to OS platform through vulnerabilities detected and verified
  - Privilege escalation if access gained is non-administrative
  - Brute force attacks on commonly known technologies
  - Cracking passwords obtained through exploitation
  - Account/Password reuse across various services
  - Networking related attacks related to Layers 2 and 3 of the OSI model

All SensePost Analysts follow the Open Source Security Testing Methodology Manual (OSSTM, which is a best-practice penetration-testing framework. Further information about the guide can be found at http://www.isecom.org/mirror/OSSTMM.3.pdf

2.3 Application Assessments

SensePost follows a strict methodology when conducting an Application Security Assessment. This ensures that a structured process is followed and provides the client with a baseline against which the quality of the assessment can be measured.

Security Assessment Methodologies

Our methodology takes into consideration industry-wide statistic projects looking at the most vulnerable areas of application deployments, including the OWASP Top 10[1] and the SANS Top 25 Most Dangerous Software Errors[2].

The specific aspects that are assessed, but not limited to, include:

System Enumeration

- Determine what the attack surface area is
- Determine what technologies are in use
- Identify input areas and other application functionality
- Understand general application function and data flow

Authentication and Authorization

- Determine what mechanisms are in place to protect user accounts and authorization schemes
- Test for known authentication and authorization flaws
- Test for user enumeration and information leakage
- Brute-force user accounts and passwords
- Test logout and browser cache management
- Test multiple-factor authentication (2FA/Certificate)
- Test forgotten password functionality and user-creation functionality
- Test for race conditions
- Test for privilege escalation

Session Management

- Analyse the session management functions implemented
- Analyse the session management token generation function for flaws
- Test session transport functionality
- Test cookie attributes

[1] https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project
[2] http://www.sans.org/top25-software-errors/

- Test for Cross-Site Request Forgery (CSRF)

Input Validation

- Test the applications ability to handle malicious input and malformed requests
- Test the input/output encoding functionality present in the application
- Test system commands in input fields
- Test for Cross-Site Scripting (Reflected/DOM/Stored)
- Test for SQL injection
- Test for LDAP/ORM/XML/SSI/XPATH/Code injection
- Test for HTTP Splitting/Smuggling
- Test AJAX functionality

Business Logic

- Determine if logic flow can be abused or bypassed

Configuration Management

- Determine if any configuration management flaws exist, such as incorrect deployment and system hardening
- Test for platform-specific vulnerabilities
- Test HTTP methods and Cross-Site Tracing

Data Encryption

- Determine what encryption mechanism is in place and the algorithms in use
- Test session cache control mechanisms
- Test SSL/TLS (SSL version, Algorithms, Key Length, Validity)

2.4 Source Code Reviews

SensePost follows a moderately strict methodology when conducting code reviews. This methodology naturally changes somewhat depending on the type of code and languages of the application being assessed; however, the basic principles remain the same.

Select the code review strategy: The following three basic strategies or hybrid approach can be used:

- **Candidate Point Approach**: This approach features 2 distinct steps. First, creating a list of potential issues through some mechanism or process. Second, examining the source code to determine the relevance of these issues.

- **Design Generalizing**: This approach is intended for analyzing potential medium- to high-level logic and design flaws.

- **Code Comprehensive**: This approach involves analyzing the source code directly to discover vulnerabilities and meanwhile improving the auditor's understanding of the application.

- Conduct automated code scans using appropriate software toolset.

- Manual verification of the automated analysis.

- **Desk Checking**: Is a technique that creates a table of all variables in a code fragment and then populates them with some initial values that the auditor thinks the code might not handle correctly. The auditor steps through each line of the function, updating each value according to the code.

- **Subsystem and Dependency Analysis**: This includes identification of string parsers, System API replacements (such as file manipulation APIs and network APIs), custom memory allocators and etc.

2.5 Wi-Fi Assessments

Wireless security assessments are conducted using a strict methodology and ensures a structured process that is followed and a yardstick to measure the assessment outcome against. The following is a more detailed discussion on the methodology followed:

Discovery

- Discovery of approved and rogue Access Points (APs), Discovery of rogue devices using existing network management implementation
- Identification of targets to be included in assessment, and
- Leakage of wireless traffic outside allowed boundaries

Wireless Device Configurations

- Inspect access control
- Identify available and vulnerable services, and
- Determine security settings

Encryption

- Assess WEP / WPA encryption, and

- Investigate additional encryption architectures

Authentication

- User Authentication
- Device Authentication, and
- Mutual Authentication

Physical Security

- Assess the physical location of the APs

In addition to the above methodologies, an internal spreadsheet exists which is completed during every assessment. This ensures we have checked, and followed, our own methodology and allows clients to request proof of this, if the situation arises.

2.6 SCADA Assessments

The testing of SCADA systems is not much different from other standard systems. However, since availability is the top priority, testing must be done with caution, especially if performed on LIVE systems. Detailed below are some of the areas we review while testing SCADA systems:

Data Security

All forms of data exchanged between all SCADA sub-systems must be protected commensurate with their criticality to the system. Data marking and need-to-know controls are important considerations.

Data Sniffing

In this test, analysts try to sniff the data exchanged between various components of SCADA systems. Sniffing could be an active or passive activity. For example, sniffing the control logic uploaded to the controller from a flex or console station.

Data Storage

In this test, analysts analyze the way in which data is stored. Data must be protected during its complete lifecycle including creation, storage and destruction. Destruction is as important as creation and storage, and it is often an adversary's easiest means of data theft. For example, is it possible for the analyst to retrieve sensitive data if it is stored on disk in un-secure manner?

Data Manipulation

In this test, analysts verify if it is possible to manipulate the system data. Also, in case the analyst is able to manipulate the data, is there a mechanism in place by which the controller can check the data integrity. For example, changing the logic uploaded from a station to the controller.

Data Sharing

In this test, the analysts analyze the way in which data is shared between different sub-systems of SCADA system. For example, in case a station and server are reading/writing the data to the same file, it is possible to create a race condition.

Platform Security Testing

Platform security testing will identify secure configuration defaults that are required within the SCADA system. The procedures for account creation and termination will be tested. Stations, servers, controllers and other devices each have a separate set of rules, which govern what entails a secure configuration. During platform security testing, we test the configurations of stations (clients), server, database server etc. This helps ensure that these sub-systems have a secure configuration and do not expose any security threat. For example, trying to create a null session with a server or trying to map the registry values, machine shares or trying to remote connect to the database and retrieving information from it are some of the kind of tests that can performed under this section.

Communication Security Testing

Communication security testing identifies the paths which data will take through a network; details protection mechanisms for different network segments; identifies security zones; and specifies external connection permissions.

Wired connectivity

This section tests the communication within the wired portions of the SCADA network, including all parts of its LAN, MAN, or WAN segments. It also tests the cryptographic implementations, if present, based on data categorizations.

Wireless connectivity

Wireless connections to a network need to have special consideration due to the broadcast nature of the medium. The security testing under this section designates what type(s) of data may traverse the wireless network, and how connections to the network are established. Also, the acceptable configurations for wireless connections to the wired network are tested. For example, does the access point cloak SSIDs, what kind of authentication is user – WEP or WPA, what is the strength of the key etc.?

Remote Access

Here we test if and how users can connect to the SCADA system from remote locations. Remote access is often a requirement in geographically large installations to effectively maintain the system. Vendors also use remote access for off-site maintenance and product upgrades. This policy details how to request access, who approves the access, and any time restrictions for the access. Once the acceptable use for remote access has been tested, we define the security controls needed for access control.

Third Party Access

This section tests if, when, and how outsiders will access information and equipment on the SCADA network. This testing tests how to request access, who approves the access, and any time restrictions for the access. The monitoring and logging is also tested. For example, various third party configuration tools open a different path to access the controllers.

Application Security Testing

Application security testing tests the various SCADA applications / software for security. Many threats and vulnerabilities arise from the insecure applications. It is very hard to block the manipulated application data through firewall since it becomes a part of legitimate traffic. An attacker can manipulate this data by various means to exploit the system.

Replay Attacks

A replay attack is a form of network attack in which a valid data transmission is maliciously or fraudulently repeated or delayed. This is carried out either by the originator or by an adversary who intercepts the data and retransmits it, possibly as part of a masquerade attack by IP packet substitution. For example, sniffing or intercepting the data exchanged between control builder and controller and replaying it back to the controller either in loop or by injecting malicious strings.

Data Integrity Attacks

This test ensures the integrity of data, which is exchanged between various sub-systems of SCADA system. It is important to ensure that the data has not been tampered while it was traveling over the wire or wireless medium. The receiving sub-system should be able to detect if the data has been tampered and should reject it.

Fuzzing

Fuzzing is a software testing technique that provides random data ("fuzz") to the inputs of a program. If the program fails (for example, by crashing, or by failing built-in code assertions), the defects can be noted. This basically forms the abuse test cases to test the secure implementation of the application. For example, a SCADA application sends an integer parameter to the controller and an attacker tries to fuzz it a long string to monitor the application behavior. If the application does not have the proper checks, it might crash. This also applies to the controller.

Authentication and Authorization

This section tests the authentication and authorization mechanism implemented in the SCADA system. An application or sub-system should authenticate itself before start communicating with the peer device or system. Also, the receiving end should be able to authorize the data and access levels allocated to the sending component.

Application Attacks

In addition to the security tests mentioned above, there are other type of application attacks like buffer overflow, format string attacks, MITM etc. which will be covered in this section.

Network Security Testing

Network security testing is a very important part of this whole activity. It will ensure that the controller and firewall has protection mechanism in place, which able to handle the malicious data packets generated by an attacker. This protection mechanism can be implemented through a firewall, i.e. by detecting and dropping the malicious data packets or can be implemented in controller itself. Ideally, it should be implemented in both the firewall and controller. This section covers attacks like Denial of Service, Distributed Denial of Service, firewall bypassing, network flooding, network scanning and fuzzing malicious data.

Through such attacks, the analyst will test the way in which controller handles malicious data. He/she will use different techniques like fragmenting data packets, stealth scanning, ping sweep, different types of scanning techniques in order to test firewall and controller. Example: Sending a packet with same host and destination IP/MAC address. This is called LAND attack.

Protocol Security Testing

This section tests the various protocols used by SCADA components and devices in order to exchange data/information with others. Protocols like DeviceNet, CNet, BackNet, Modbus etc. are widely used in SCADA systems and thus it is important to ensure the security of communication happening through these protocols. This section widely covers protocol manipulation, protocol fuzzing where various parameters will be fuzzed in order to test the robustness of the protocol. For example, trying to send a Modbus command to a device without proper authentication/authorization. There are commands, which can flip the logic, reset the device status, re-start the device etc.

HMI Security Testing

Involves testing of the operator interface and the underlying framework (such as step7, wonderware, etc) for the following:

- Escaping from HMI to the operating system
- Unauthorized access to menus and options
- OS command execution
- Interface manipulation