Module: Unauthorized Access

Panos Papadimitratos
Networked Systems Security Group
www.ee.kth.se/nss

November 16, 2015

Contents

1 Introduction (Slide 1-4)

1.1 Basic and broad definitions for Unauthorized Access (Slides 3-4)

Unauthorized Access is the act of illegal access to resources, network infrastructure or computers and databases without permission. An example of an intrusion intrusion action is to steal (or guess) an administrator's password, in order to gain access to a server with root privileges [1].

1.2 Intruders (Slides 5-10)

Intruders can be outsiders or insiders. Common intrusion techniques include: password brute-forcing, dictionary attacks, exposing known system's vulnerabilities, bypassing firewall rules or a biometrics authentication system. Intruder targets may be user records stored in server databases, or email accounts. The impact of the intrusion can vary according to the privileges that the intruder has, her/his knowledge, ability to hide her/his tracks and the importance of target.

Intruder Classes (slide 6) A classification in three general intruder classes:

Misfeasor: a legitimate part of the system who misuses his privileges to access critical parts of the network that he wasn't authorized to. The misfeasor goes through access control holding legitimate credentials but takes advantage of system vulnerabilities to access more services and resources than those s/he is supposed to.

Masquerader: an individual who gains access to the network without being authorized to do so (also called outsider). For example a masquerader may access an email account without being authorized to. They usually exploit legitimate users' passwords, to pass through access control. Brute-forcing and guessing popular passwords are common techniques for the masqueraders.(e.g. "123456", "username" or "mother's name")

Clandestine User: gains administrative access to the network (or is already authorized as an administrator) to overpower any auditing countermeasures and uses them with illegal motives.

Intruder Patterns (slides 7-9) Intruders can be further categorized according to three patterns, notably based on motives, power, and technical knowledge (the latter two also depending on the resources/funding) to launch the attack.

Hackers: they may hack for thrill and fun. Their targets are usually small companies and individuals. Therefore, their hacking results have less impact or publicity, but can still be disastrous. The hacking community likes to share new vulnerabilities with its members and also publishes hacking results online

[1] The introduction is based on "Network Security Essentials, fourth edition, chapter 9" by William Stallings

e.g. lists of stolen passwords and usernames. For *metasploit.com* supports an online community where vulnerabilities and exploits are shared. Such online information can be useful both for hackers, but also for security engineers who try to solve the problems that have been recently discovered. Also, there is also a plethora of available tools for system penetration testing, passwords strength checking, packet capturing etc. Some examples are *nmap, john the ripper, hydra and wireshark*.

Criminal Enterprises: Organized groups that launch sophisticated attacks. Criminal enterprise attacks are becoming more and more popular. Targets are usually bigger and include companies, banking institutions. Country facilities and institutions have also been multiply targeted in the past by such groups (stuxnet). The size of their targets and the sophistication of the attack, indicate that criminals are well funded and supported by people, corporations, countries and they could be as big as their targets.

Insiders: already part of the system, with valid credentials and access privileges, and they are extremely hard to detect. Their motives can vary, ranging from "whistle-blowers" to disgruntled organization members to attackers that leveraged social engineering methods. Recent prominent example of an insider? If you have not seen it yet, Citizenfour could be quite interesting.

2 Firewalls (Slide 11-20)

Firewalls are hardware or software programs that control the flow of network traffic between hosts or networks. They are a vital towards protecting networked resources, performing packet level inspection from different layers of the OSI model. They are placed in front of the network (or host) they protect, providing a first line of defence. Firewalls can be categorized according to their operation/functionality:[2]

2.1 Packet Filters (slide 13)

Packet filters operate at the network layer (layer 3 of the OSI model) and apply a set of rules to incoming and outgoing IP packets. They can check source and destination IP addresses or the protocol used for communications. Some more advanced packet filters can also check characteristics of the transport layer (layer 4 of the OSI model), such as source and destination ports. Packet filters are the simplest form of firewalls and most modern firewalls also perform packet filtering. A list attributes that packet filters check are:

- Source IP address
- Destination IP address
- Source and destination transport-level address: port numbers that can define applications e.g. TELNET port 23

[2]National Institute of Standards and Technology (NIST):`csrc.nist.gov/publications/nistpubs/800-41-Rev1/sp800-41-rev1.pdf`

- IP protocol field: defines transport layer protocol
- Interface: for firewalls with three or more connections ('ports'), define where the packet came from or which one it is destined for

If an incoming packet finds no match to any of the rules defined by the packet filter, then two of the default policy actions can be taken:

- discard: if not explicitly accepted then it is rejected
- forward: if not explicitly rejected then it is accepted

The "discard" policy can be handy when a more conservative policy is required, e.g., for corporate environments. Packet filters should also consider the IP addresses of outgoing packets. Source addresses of outgoing packets other than that of the network behind the firewall, may indicate active attacks against other networks. Recall the discussion regarding filtering reg. DDoS attacks.

When your firewall rejects a packet, does it notify the source of the packet? It could, but it is advised not to; if it did, it would reveal sensitive information to the attacker - leaking information on what are the implemented rules and essentially the policy.

Weaknesses (Limitations) (slide 14) Packet filters are simple and easy to implement. They can allow allow or deny access according to the destination IP addresses. Unfortunately, IP addresses can be spoofed, which means that an adversary can change his IP address to an accepted one that is not predicted by the firewall administrator and thus, pass through the firewall. As an example consider an SSH client connecting to an SSH server that is protected with a firewall not accepting connections from the network 240.237.50.0/24 . If the attacker spoofs his IP address and changes it, e.g., to 241.238.40.28, then the packet filter will let the connection through and completely ignore the application running (whether it is SSH, and instant messaging (IM) protocol etc).

Source routing attacks is another vulnerable point for packet filters. Source routing specifies and attaches the route for the packet. This way an attacker can get packets to reach a machine that would not normally be reachable from outside the network. An obvious countermeasure is to ignore source-routed packets. Another attack is IP fragmentation: the attacker cuts the original IP packets into very small fragments, and forces the TCP header information into a separate fragment. This way the attacker hopes to trick and bypass rules that rely on transport layer information. The countermeasure to this attack is to request at least a minimum amount of TCP header information to be included in the first IP fragmented packet that is the one to be inspected. Finally, no validation of data coming through is performed.

2.2 Stateful Inspection Firewalls (slides 15-16)

Stateful Inspection firewalls offer improved security compared to packet filters. They still inspect IP packets but they also perform a transport layer check to the packets. Stateful inspection firewalls perform the additional task of monitoring the state of the current connections (sessions) to state tables. An example of a state table is given in the slides. Connection status can be connection

establishment, usage and termination. An attacker would have to create packets similar to an active connection and hope that they will pass through the firewall. Moreover, stateful inspection firewalls also check TCP packet sequence numbers, which could probably detect the type of attack described above. Obviously if the UDP protocol is used no information regarding the status of the connection can be kept, since UDP is connectionless.

To understand how security is improved it should be reminded that when two hosts communicate they assign a port for that specific connection. For the classical client-server application, a standard port in the range of 0 to 1023 is assigned to the application at the server side. On the client side a random port can be assigned to the application during the communication. Consider the SMTP protocol (Simple Mail Transfer Protocol) that uses the TCP transport layer protocol for communications. As explained above, the SMTP client is randomly assigned a port in the range of 1024 to 65535. A packet filter firewall would not keep track of this connection and would allow all incoming TCP connections traffic for the above port range. On the contrary, a stateful inspection firewall would keep its status in a state table, inspect source and destinations ports, IP addresses and connection status. All incoming and outgoing traffic should comply with one of the entries of the state table. Therefore stateful inspection firewalls can prevent a larger set of attempted unauthorized access that could go unnoticed by a typical packet filter.

Note also that fiewalls can count packets of a given type, or enforce specific rates that are acceptable. For example, connect this to the smurf attack in the previous lecture: without precluding/forgoing ICMP echo requests, the firewall can allow an acceptable rate of such packets, on the one hand limiting the scope/impact of the attack and on the other hand maintaining the functionality.

Moreover, note that the expressiveness of these rules is not arbitrary - it relates to the considered protocols and their functionality. More elaborate patterns, or attacker behavior, can be captured by intrusion detection systems discussed later.

2.3 Application Firewalls (slide 17)

Application firewalls check packets based on information from the application layer. This means that they can check how the application runs over the network. Legitimate user profiles are compared to malicious behaviour patterns. For example a malicious behaviour can be the execution of an Instant Messenger protocol over port 80 (intended for http use).

2.4 Application Proxy-Gateways (slides 18-19)

Application Proxy Gateways are advanced modern firewalls that provide lower layer access control to the network, as well as upper layer functionality. They usually serve as the front end to connect two different and distant networks or hosts over the internet. They relate directly to or apply in Virtual Private Networks. Application Proxy Gateways operate at the application layer and they inspect the content of the data exchanged between the communicating parties.

They also serve as agent-intermediaries between two communicating networks, remaining transparent to them. This means that when a host-to-host

communication link is established, there will consequently be two underlying connections, one between the client and the proxy agent, and another between the proxy agent and the destination. If the destination is also protected by a proxy gateway, then it is clear that there exist three underlying connections.

Connections passing through the proxy agent must comply with the firewall's ruleset. Except from the firewall ruleset, the proxy can require user authentication in terms of passwords, certificates etc. Application Proxy Gateways offer additional layers of security compared to the rest types of firewalls. They can support decryption and re-encryption of packets before forwarding them to their actual destination. Moreover, packets are checked at the proxy, which is separate from the destination, preventing direct communication between two hosts. Obviously improved security comes with a cost in efficiency.

An intruder would have to authenticate at the proxy before reaching the destination, which means it should have stolen credentials (passwords, certificates) or be previously admitted to the system through a legitimate process (for example, . Furthermore, intruder actions should conform with the ruleset of the firewall, at all the application, transport and network layers.

3 Intrusion Detection Systems (Slides 21-30)

Intrusion Detection Systems (IDS) are used to detect adversarial behaviour and reveal attacks against the system. Adversaries can be identified by keeping logs, i.e., collecting data of legitimate user behaviour; or else by specifying what adversarial behaviour is. IDS systems can be divided in two categories: anomaly detection and signature-misuse detection.[3] [4] [5]

3.1 Anomaly Detection (slides 23-24)

Anomaly based detection techniques create profiles of correct user behaviour. Data related to legitimate users is collected to create the profiles. Deviations from expected user behaviour and abnormality is therefore considered suspicious. Creating successful user profiles and avoiding false positive alarms is a hard process. Examples of adversarial behaviour many unsuccessful login attempts within a short period of time or multiple attempts to log in with administrative credentials. Models of correct behaviour are created using two general approaches:

Programmed models Programmed correct behaviour can be specified by the administrator who creates statistical models based on experience and expected operations, such as number megabytes of files transferred per day by a typical user.

[3] National Institute of Standards and Technology (NIST): `csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf`

[4] Intrusion Detection: A Brief History and Overview `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1012428`

[5] Intrusion Detection Systems: A Survey and Taxonomy `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.1.6603`

Dynamic models Dynamic models are based on self-learning techniques. These are of stochastic nature and observe users in time. They have the advantage of detecting unknown previous attacks, which would otherwise have to be explicitly specified by the administrator in programmed approaches.

Can the adversary take advantage of the dynamicity of the system and learning component? It could possibly influence the learning approach. But let's consider the definition of a 'normal' situation: there is always a margin to cater to dynamically changing conditions. For example, packets being dropped at some low rate could the norm; setting a threshold pretty slim could result in false positives when there is some transient impairment. Setting the threshold more generous avoids such false positives but it allows the adversary to drop just enough packets while remaining below the 'radar'/threshold. Not devastating but still reducing the effectiveness of the system. Of course, the attacker could be well-behaved for as long as it wishes to (being undetected or in that sense indistinguishable from a benign node), and then begin a full fledged attack, which would be debilitating but detected.

3.2 Signature Detection (slide 25)

In signature detection (commonly referred to as (rule-based) misuse detection), the intruder behaviour is specified and not the correct user behaviour. In other words, the IDS does not looking for deviations from correct behaviour, but directly for pre-defined intruder actions. Despite having less false positives, it is hard these techniques capture all intruder profiles.

3.3 Honey Pots (slides 26-28)

The objective of honey pots is to deceive an intruder and lure him to attack the honey pot, instead of any other network's infrastructure. To achieve this, honey pots seem to have several security vulnerabilities that the intruder can exploit. By trapping the intruder in the honey pot, the system can monitor behaviour and check the intruder's capabilities or objectives. In order to avoid detection of honey pots, they should be kept in generic form e.g. unmodified systems with popular vulnerabilities. Another point to consider, is that the honey pot should not become a starting point of attack to other networks e.g. to launch Denial of Service attacks as part of a botnet.

4 Authentication (slides 32-67)

Given the importance of authentication, to further protect networked resources, the rest of the lecture revisits authentication techniques. In particular, we discuss access through passwords (a single one per user) for a set of networked resources, public/private key based authentication, and virtual private networks (VPNs). We will revisit IPsec, relating to VPNs, in a subsequent module.

4.1 Passwords (Slides 35-61)

Passwords are hidden codes held secret by users, and presented in order to authenticate themselves. There are two main attacks against passwords:

brute force attacks (slide 36) The attacker tries every key combination in the worst case, trying to find the correct key to decrypt the cipher.

dictionary attacks (slide 36) The attacker holds a long words list (the dictionary) which are the most common passwords used. Despite their simplicity, dictionary attacks have proven to be extremely successful as users tend to use popular passwords and user names (e.g. "123456").

Passwords (slides 37-38) Passwords should be hard to guess and and also difficult to brute force. Today, there are distributed computing systems and supercomputers that can perform 1.000.000.000 computations per second. Such powerful infrastructure may be in the hands of criminal enterprise groups. Weak passwords can be cracked immediately by such machines. Past experience and former security breaches (see RSA breach) have taught us that "system security is often as strong as the weakest password".

Password Strength (slides 39-40) Entropy is a measurement of unpredictability and variance. Password strength can be expressed using entropy, meaning how difficult it is for the attacker to brute force it according to the variety of symbols used (the alphabet). The mathematical formula to compute entropy H is the following: $H = Llog_2(N)$, where N is the size of the alphabet used and L is the length of the password. In general terms, symbol variety and bigger passwords combined, are harder to break. Nowadays the use of lowercase, uppercase, digit and special characters symbols is strongly encouraged.

Secure Password Storage (slide 41-42) Secure password storage is a crucial point for the security of a system. Storing passwords in plain-text is a very bad security decision, because the database should not be considered as safe against adversarial attacks. Storing passwords hashes instead is another bad tactic that adversaries can take advantage of and retrieve the original plain-text password values. Rainbow Tables that assign common passwords to hashed values can be used by the adversaries to retrieve the passwords. The technique is simple and relies on common patterns that most users use in their passwords. The solution to the problem is to use random values called salt that are hashed with the plain-text password before it is stored. Adding this randomness makes the work of adversaries much more difficult to retrieve the original passwords, even if they have access to the database.

4.2 One Time Passwords (Slides 57-61)

One time passwords are an enhancement to the classical password security. They are valid for one session or transaction only. One time passwords cannot be reused and they find many applications in online banking systems. A common technique for one time passwords include a secret pass (PIN code), hardware (cards and small crypto devices) and additional information which is either public (social security number) or obtained at the time of logging (website generated code). Swedish banks use one time passwords to authenticate their

users (e.g. Nordea Bank)[6].

(slide 57-58) Each user is provided with a smart card, a pin number and a small crypto device. The user inserts his card to the crypto device, fills in his social security number to an online form, and inserts a code provided by the bank's website in the device. Finally he inserts his secret four digit PIN number and obtains the one time password he can use to access his account. The whole session is encrypted using https to defend against eavesdroppers and to assure that the user is connected to the correct bank's server.

(slide 59) It would be an extremely difficult task for intruders to directly attack a bank's website. Similar intrusion attacks have happened in the past, but require technical knowledge, sophisticated attack plans and perhaps expensive infrastructure. At least one of the above parameters are missing from a typical group of intruders. The most convenient way would be to find a technique to obtain the one-time passwords of legitimate users and use them. To achieve this the attackers can use a phishing attack. Phishing attacks are attempting to lure typical users to reveal personal information useful to mount an intrusion attack e.g. passwords. Typically the user will be redirected to a website controlled by the adversary that will have the exact same looks as the original one.

Attack against Nordea (slides 60-61) Russian attackers contacted several Nordea's customers pretending to be the bank's representative and asked them to download anti-spamming software. The unsuspecting clients downloaded and installed a trojan horse instead. This small piece of software would be activated each time the client would try to login to Nordea's website. An error message was being displayed and the users were being redirected to a phishing site. The users would then give in their passwords to servers controlled by the attackers. Using phishing sites the attackers gained access to many online accounts. Recall that social security numbers also required to access the sites are publicly available and not hard to obtain.

4.3 Kerberos (slides 45-56)

In order to minimize the cost of user authentication in a local area networks (compared to PKI implementation), the Kerberos protocol can be used instead. Kerberos was developed in MIT in the mid 1980s as a method of user authentication using passwords over untrusted networks, but trusted hosts. This assumes that the network traffic can be captured by an eavesdropper, but the computers used to by the user to log on, and the software running on them, are trusted. Currently most implementations use Kerberos version 5 and only a few Kerberos version 4 implementations still exist.

Each user is provided with a secret password used to log on and request for a network service. Users passwords are also saved in a secured central database. A trusted server called the Authentication Server (AS) has access to these passwords and is used to authenticate users. In an overview of the system the user U obtains a ticket from AS that proves her identity. Then, using the ticket

[6]Case Study: Online Banking Security `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1621055&tag=1`

she requests for Service Tickets from a Ticket Granting Server (TGS), without having to re-enter passwords. The last tickets can be presented to the service provider and gain access to the service. A more detailed analysis of the Kerberos system follows:

step 1 (slide 52-55) U first sends a ticket request to AS revealing her identity (U_{id}), but the assigned password K. Since AS has access to the secured database where K is stored, it already knows the password for that user. Therefore it generates a secret session key K_c from K. AS then replies with a message encrypted with K_c containing a *ticket*. The ticket is an encrypted reusable proof of authentication of U that can be used multiple times. The ticket is an encrypted message with a different secret key that shared between AS and TGS. The ticket also included the id of U for whom the ticket was issued. Therefore, the user (or any eavesdropper) cannot change any values of the ticket, for example its lifetime. In the ticket, yet another special session key is also included, the K_{uTGS}, which is used for communications between U and TGS. In the next step, U will send an encrypted message to TGS with K_{uTGS}. K_{uTGS} is also included in message encrypted with K_c (meaning also outside the ticket) so that U can find it out and proceed to the next step.

Note that at this point an intruder who captures the first message cannot extract useful information. He cannot decrypt the reply message from AS. Only the correct user can also create K_c, decrypt the message and therefore obtain the ticket. Each ticket is assigned a lifetime parameter by the AS, so that it expires after some hours have passed (10 by default). The lifetime of the ticket is included in the encrypted part and cannot be changed. This allows the user to ask for additional services without re-entering his password, but also gives an intruder the chance to reuse the ticket in case he captures the working station from where the legitimate user logged on.

step 2 (slide 56) Having the ticket and a session key to use for further communications with the TGS, the user approaches TGS. The TGS server issues tickets for specific network services. U sends his ticket and an authenticator message which is encrypted with the session key K_{uTGS}. Recall that the session key is also included in the ticket and this means that an intruder cannot fabricate its own session keys (the authenticity of the ticket can be checked).

The ticket can be decrypted by TGS, because it is encrypted with a secret shared key by the AS and the TGS. The session key included is then revealed to the TGS. Then the the authenticator is decrypted and if information included matches that of the ticket and is valid (e.g. ticket not expired), the user is issued another service ticket.

step 3 (slide 57) The TGS server then sends a message following the same format as that for the AS reply to the U. A ticket is included in the message containing the session key between the user and the service provider.

step 4 (slide 58) The User, U, can finally be authenticated to the service provider the same way that he was authenticated to the TGS.

4.4 Multi-Realm Kerberos (slide 60)

A Kerberos realm is the architecture described in the previous slides. It consists of the managed nodes sharing the same passwords database. Networks of clients and servers consist different realms. Multiple single realm Kerberos systems can coordinate and create an inter-realm Kerberos environment. The need to have multi-realm Kerberos systems comes from the fact that users may want to access services offered by other Kerberos realms.

To achieve cross-realm authentication for users each Kerberos server should share a secret key with the servers in the other realms. This implies the trust of a Kerberos server to another server to authenticate its users. The authentication scheme remains the same, but at step 3, instead of requesting access to a local service; the user requests access to the distant service provider. The ticket presented to the remote Kerberos server (TGS remote) is encrypted with the shared key between the Kerberos servers.

4.5 Kerberos Overview (slide 61)

Kerberos assumes that hosts and software running on hosts that are used by users to log on, are trusted. This obviously is not the case for real scenarios. Password based security is a weakness point also as it has been already shown. On the other hand, Kerberos 5 improves the security of Kerberos 4 in many fronts, such as using stronger cryptographic primitives. Moreover, Kerberos is suitable for local area and neighbouring networks. For authentication of distant, mobile users over the internet, PKIs and digital certificates are needed. An old but good overview of Kerberos can be found at the link of the footnote [7].

4.6 Public Key Infrastructures (slides 62-69)

In order to authenticate hosts, vendors, banks etc over public and open networks a more reliable and secure system than simple password authentication is needed. Public key Infrastructure (PKI) provides a way of authentication based on public key cryptography and digital certificates. PKIs are being extensively used in the Internet of Things and especially in e-commerce nowadays. A PKI is defined as the set of hardware, software, people, policies and procedures involved to manage, store, distribute and revoke digital certificates. Note that we will keep revisiting teh matter, considering PKIs for different purposes across different modules.

The goal of the PKI is to bind an identity, say an e-vendor, with a public key that corresponds to a private key known only to the vendor. When someone wants to contact the vendor securely, she can use the public key. In order to know which is the correct corresponding key and that no malicious third party can read the exchanged information, public keys are digitally signed by Certification Authorities. Since everyone trusts the CAs, they will trust the certificates signed by them, and thus the vendor.

[7] Kerberos: An Authentication Service for Open Network Systems `http://www.cse.nd.edu/~dthain/courses/cse598z/fall2004/papers/.pdf`

The Certification Authority (CA) issues, revokes and manages certificates obtained by vendors, clients etc. The role of identifying if the entity requesting for a certificate is real, is usually taken by a Registration Authority (RA), an intermediate between the CA and the certificate requester. The RA identifies the entity requesting for the certificate by asking for identities, e-statements etc. Then and if the identification is correct, it issues a request for certificate from the CA.

A digital certificate is an electronic document and contains more information than just the public key. It contains information about the issuer of the certificate (signer of the certificate), the holder of the certificate, the validity period of the certificate the corresponding public key for the certificate holder etc. The most common format used for digital signatures is the X.509. PKIX is a working group established in 1995 with the goal of developing Internet standards to support X.509-based Public Key Infrastructures (PKIs). SSL certificates (used in SSL/TLS) follow the X.509 format also and are used widely over internet communications. They are used to authenticate your bank's website for example. Your browser has a set of root keys pre-installed, meaning public keys of CAs.

Problems with PKIs PKIs offer flexibility and have provided a way to secure communications, but still there are some open issues and risks. The first big question regarding CAs is who we trust and for what reason? Who decided which CAs are to be trusted and who could be a CA? Another important question is "what is certified?". Is the CA responsible to certify DNS domains (included in certificates) or individuals, vendors etc. Should DNS domains be certified by specific domain CAs? Moreover, private keys are stored locally on servers and computers. If not stored safely they could be used to produce certified digital signatures by anyone who has access to them. Therefore the old problem of secure password storage is not solved. Moreover, there are still issues with identifying the holder of the certificates. Is binding a public key to an identity enough, and if so, how does the CA know about the identity? Does the RA solve this problem? Finally, if the root certificates list is exposed, say to intruders, then any entry added to that list is still identified as trusted. It should be noted here that these problems do not diminish the importance and usefulness that PKIs play for today's networks. There are still open problems though and alternative solutions or stringent policies are under study and argumentation. A more detailed analysis of the risks of PKIs can be found at the footnote[8].

The Comodo Breach The attack against Comodo brought into attention that there are still things to consider about PKI security. Comodo is one of the major digital certificates issuers. According to the reports, an RA affiliated with the Comodo CA was infiltrated, when an administrative password was stolen. The intruders then requested for certificates for various domains, including "mail.google" and "yahoo.login". The attack was identified and the fake certificates were revoked.

[8]Schneier on Security: `http://www.schneier.com/paper-pki.html`

4.7 Smartcard Authentication (slides 70-73)

Smart cards are minimized versions of computing systems used for authentication purposes. They include a weak CPU, RAM/ROM memory and a minimal OS version stored in the ROM. Cryptographic keys can be stored securely in the EEROM, in the sense that they cannot be retrieved by someone who steals the smartcard. Smartcard authentication has several benefits over conventional authentication methods like passwords. Several keys can be stored in the smartcard which increases the efficiency and mobility. Furthermore, weak passwords and/or lost or old forgotten passwords are no longer a threat. Of course this is not the case with a lost smartcard. To avoid the danger of an intruder using a stolen smartcard two methods are used to activate it. The first method is to use a secret PIN (usually 4 digits long) and the second method is to use biometrics. The later choice could be considered safer and could perhaps be the case in several years from now, since smartcard usage is already envisioned as a promising online authentication scheme for future e-commerce and not only.

GSM (Global System for Mobile Communications or Groupe Spécial Mobile) uses SIM (Subscriber Identity Modules) cards to authenticate users to the network. When a user wishes to access the network, she is challenged with a challenge R sent to her mobile. The sim card has a 128-bit long key K_c stored and using algorithm $A3$ computes a 32-bit reply called $SRES$. $SRES$ is sent back to the authentication server. $A3$ and K_c are also stored at the server so the $SRES$ value can be computed and compared to the reply. Both parties then use the $A8$ algorithm to compute a 64-bit ciphering key K using $SRES$ and K_c as inputs. The K_c is the ciphering key that is used in the $A5$ encryption algorithm to encrypt and decrypt the data that is transmitted.

4.8 Biometrics Authentication (slides 74-79)

In biometrics authentication the user's identity is the verification of the user by what he is. This means that physical traits and biological data (iris) are verified to authenticate the user. Identity theft in such systems is much more difficult, but still not impossible. For example consider the case of eye contacts with the same iris pattern of a legitimate user.

The authentication process starts by capturing the data, e.g., getting fingerprints, scanning the iris, using special devices. The data captured then go through signal processing to translate them into a format, comparable to database records. If the captured data match those of the database then the user is authenticated. If not he is rejected from the system. Errors in authentication using biometrics may exist, because captured data can variate according the user's psychology and mental condition etc. Moreover physical traits may change over time. For this reason, there exists a special mechanism monitoring data change over time and refreshing records in the database.

5 Virtual Private Networks (slides 80-92)

Virtual Private networks provide a virtual and secure link connecting distant networks. Examples are company branches in different countries and corporate networks, or remote employees that want to access their headquarter's

servers. The term virtual comes from the fact that data are tunnelled through the internet, emulating a point-to-point connection. This means that the whole tunnelling procedure is transparent to the user. Network-to-network communication is performed via proxy gateways that share Security Associations (SA) and form secure tunnels over these SAs. Remember that a Security Association is the set of agreed security attributes to use between two communicating entities that use IPSec. SAs are established using the Internet Key Exchange (IKE) protocol, which considers X.509 certificates and Diffie-Hellman key exchange to establish the shared secret keys. Therefore the two entities (e.g. gateways or the hosts) can verify that they communicate with the legitimate party they are supposed to and can also securely establish shared keys.Secure tunnelling provides confidentiality, integrity, authentication and access control. The term private comes from the secure tunnelling implementation. The main advantage of VPNs is their low cost of deployment.

VPN security raises the bar for intruders, using IPSec or SSL for standard client/server applications over a web browser. IPSec can guarantee data confidentiality and integrity of each IP packet, according to the protocol used (AH or ESP). Recall that AH offers IP packet authentication and ESP data encryption and optionally authentication. In the basic form of secure tunnelling, each IP packet is encapsulated e.g. at the source gateway and de-encapsulated at the destination gateway.

5.1 Tunnelling in VPNs

In single tunnelling (slide 77), security is provided between gateways and no hosts implement IPSec. Each gateway implements a secure tunnel SA with the next gateway. In slide 78 a multi-layered tunnelling is presented. In this case a host-to-host SA also exists. Each upper (or outer) layer cannot decrypt and obtain what is protected in the inner layers of the tunnel. Gateway-to-gateway tunnelling provides security for traffic between the networks. Data exchanged between the hosts remain confidential to the rest of the world. Recall that in ESP tunnelling mode, whole IP datagrams are encapsulated within encrypted shells for each new secure tunnelling association. In AH entire IP packets are encapsulated inside another. In both protocols, in order to access lower layers or authenticated/encrypted IP packets a SA must be established with the sender.

6 Examples of Documented Attacks

The RSA breach: RSA is a security company and one of the top providers of security solutions for businesses. They offer authentication, encryption and access control services among others. The SecureID mechanism is an authentication method of users, using two factor authentication. The mechanism consists of a token (sth the user has) and a four digit PIN (sth the user knows). The token has an internal clock which is used to produce different token codes every 60 seconds. The PIN together with the token code consists the user's passcode, valid for 60 seconds. When the user wants to authenticate herself, she gives the combination of her identifier (similar to username) and pass code. The authentication server then computes the passcode and compares the given one with the computed one. If they match then the user is authenticated. In 2011, RSA fell

victim of an extremely sophisticated and organized attack, a typical example of the criminal enterprise type of intruders. According to media and reports, algorithms creating the token codes, users' PINs, and information linking specific tokens to users were stolen.

Description of the attack: The type of the launched intrusion attack is referred to as Advanced Persistent Threats (APTs). These types of attacks occur in three steps. In the initial step of the attack, a social analysis attack of employees and people involved with the victim network are studied. The goal of this step is to find methods to approach the employees. Then a phishing email is usually sent to some of the employees that are considered as more vulnerable first targets. In the content of the email there is usually an attached file containing a trojan horse, worm etc. Once inside the network with the infected working station, the next step is to seek for administrative accounts and passwords. The infected workstation is usually controlled by distance. Administrative passwords can be obtained through interactions of the infected account and his administrators e.g. yet another trojan. Once the intruders have administrative passwords and access to the resources they are interested in, for example a database, the quickest part of the attack follows, which is stealing the data. The whole attack can last for a long period of time.

APT attacks: The attack against RSA was an APT. The procedure followed was as described above. The initial intrusion happen when a single employee retrieved the phishing email from the junk folder of his email client and opened it, thus executing the trojan. The file had the catchy tittle "2011 Recruitment Plan.xls". As a result of this attack more followed against companies and organizations that used the SecureID system, like the attempt to attack Lockheed Martin.

Stuxnet: Stuxnet was a computer worm designed to attack the Iranian nuclear plants and was discover in June 2010. The real impact of the this intrusion attack, was that once inside the network, stuxnet could gain physical access over infrastructure. It could execute its own code into Programmable Logic Controllers (PLCs) and define the way that critical parts of the nuclear plant would operate. Duqu is considered as the successor of stuxnet that aims at gathering intelligence and sending it back to the attacker. A video of how the stuxnet worm works can be found at the footnotes [9].

NASA hacking attack: NASA discovered in November that intruders using IP address in China broke into the network of NASA's Jet Propulsion Laboratory (JPL). 23 spacecraft conducting active space missions, including missions to Jupiter, Mars and Saturn are managed by JPL. According to the reports, "intruders gained full system access, which allowed them to modify, copy, or delete sensitive files, create new user accounts and upload hacking tools to steal user credentials and compromise other NASA systems. They were also able to modify system logs to conceal their actions. ...intruders had compromised the

[9]Stuxnet video from Symantec: `http://www.symantec.com/tv/products/details.jsp?vid=673432595001`

accounts of the most privileged JPL users, giving the intruders access to most of JPL's networks" [10].

[10]Reuters on NASA hack:http://www.reuters.com/article/2012/03/03/us-nasa-cyberattack-idUSTRE8211G320120303; NASA Cybersecurity: An Examination of the Agency's Information Security http://bit.ly/yQFSB8