System Description and Risk Analysis

Marc Gähwiler Leonhard Helminger Fabian Zeindler

13.10.2013

Contents

1 System Characterization

1.1 System Overview

The system's main mission is to provide a Public Key Infrastructure (PKI) for the fictional company "iMovies". Each user (authenticated by his credentials, which are stored in a legacy MySQL database or his already created private key) can change his user information (first and last name, his email address and his password), create and revoke certificates and download the private key matching to his created certificates. To achieve this goal it implements a Certificate Authority and the following user interfaces:

User Web Interface Simple interface to change credentials, create/revoke certificates

Admin Panel Dedicated interface to see the current CA's state

Key Backup Backup of every signed certificate and the according private key

System Administration Administrators have SSH access to every server

1.2 System Functionality

1.2.1 User Interface

A simple web interface which allows each user to log in either with his credentials from the legacy MySQL database, or one of his previously generated certificate and private key combinations. Once logged in the user can view his information (last name, first name and email address), change his password and update his information (last name, first name and email address). Additionally it is possible for the user to let the system issue a new certificate (based on his possibly changed credentials) and download the certificate with the newly generated private key in PKCS#12 format.

1.2.2 Administration Interface

A simple web interface (not the same as the user web interface) where CA administrators can consult the current CA state after a log in process which requires the CA administrators to authenticate themselves with their certificate. This includes the number of issued certificates, the number of revoked certificates and the current serial number.

1.2.3 Certificate Issuing and Revocation

The CA offers an interface that allows other systems to

- Generate new public/private key pairs

- Generate a certificate that ties a public key to a user's credentials (first and last names and his email address) and sign this certificate with the CA's key
- Revoke a previously signed certificate
- Get a list that contains all revoked certificates (Certificate revocation list)

1.2.4 Key Backup

To prevent the loss of any information, that was encrypted with an issued certificate, every issued certificate and the according private key are archived.

1.2.5 System Administration and Maintenance

Each server is remotely accessible per SSH. . . . (Log in only with a SSH key, no password authentication, not accessible from outside the LAN/port forwarding or something like that)

1.3 Components and Subsystems

1.3.1 Platforms

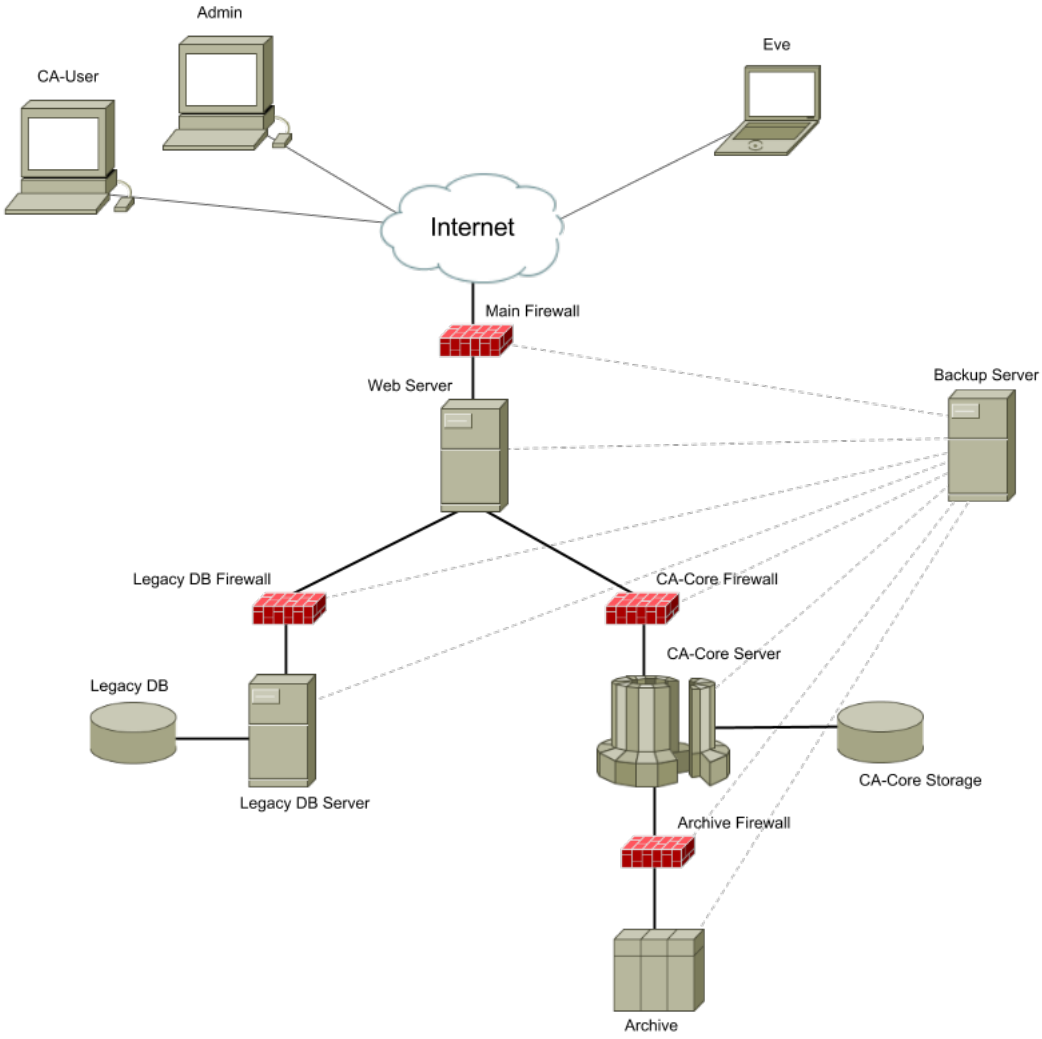

Figure 1: Network diagram

The components (as depicted in figure **??**) are as follows:

Main Firewall A virtual machine running a software firewall in front of the whole internal network. It forwards TCP Packets on port 80 and port 443 to the web server, TCP packets on port 2211 to the web server, TCP packets on port 2220 and 2221 to the legacy db firewall, TCP packets on

port 2230 and 2231 to the CA-Core firewall and TCP packets on port 2240 and 2241 to the archive firewall. It accepts TCP packets on port 2200 which is the port, the internal SSH daemon is bound to.

Web Server A virtual machine running ... linux. It only accepts connections from the main firewall. The only running daemons on this server are a SSH daemon listening on port 2211 to administrate the server and a HTTP daemon listening on ports 80 and 443.

Legacy DB Firewall A virtual machine running a software firewall in front of the legacy database server. It forwards TCP connections on port 2221 and TCP connections from the web server on port 3306 to the legacy database. Additionally a SSH daemon is running on port 2220 to allow administrators to remotely administrate the firewall.

Legacy DB Server A virtual machine running ... linux. It only accepts connections from the legacy db firewall. The only running daemons on this server are a SSH daemon accepting connections on port 2221 to administrate the server and a MySQL daemon on port 3306.

CA-Core Firewall A virtual machine running a software firewall in front of the CA-Core server. It forwards TCP connections on port 2231 and TCP connections from the web server on port 443 to the CA-Core. Additionally a SSH daemon is running on port 2230 to allow administrators to remotely administrate the firewall.

CA-Core Server A virtual machine running ... linux. It only accepts connections from the CA-Core firewall. The only running daemons on this server are a SSH daemon listening on port 2231 to administrate the server and a not yet decided service listening on port 443.

Archive Firewall A virtual machine running a software firewall in front of the CA-Core server. It forwards TCP connections on port 2241 and TCP connections from the CA-Core on port 443 to the CA-Core Storage. Additionally a SSH daemon is running on port 2240 to allow administrators to remotely administrate the firewall.

Archive Server A virtual machine running ... linux. It only accepts connections from the Archive firewall. The only running daemons on this server are a SSH daemon listening on port 2241 to administrate the server and a not yet decided service listening on port 443.

Backup Server A virtual maching running ... linux. It accepts connections from the main firewall, the web server, the ca-core, the legacy db and the archive. A SSH daemon is listening on port 2250 to allow administrators to remotely administrate the server and allow the other servers to backup their configuration files and their log files.

1.3.2 Applications

User Web Interface Web application, details not yet decided. Running on the web server.

Administration Panel Web application, details not yet decided. Running on the web server.

Legacy DB MySQL database with the legacy scheme. Running on the legacy DB server.

CA-Core Application using the OpenSSL library that provides basic interfaces to create new key pairs, sign existing key pairs and revoke certificates. Running on the CA-Core server.

CA-Core Storage A database that is used by the CA-Core to store certain data (details are t.b.d.). Running on the CA-Core server.

Archive Details not yet decided. Running on the archive server.

Backup Simple script that keeps multiple backups from the other servers (especially of all the configuration and log files). Running on the backup server.

1.3.3 Data

User Information Basic information according to the schema of the legacy database. This includes the user's username, his first and last names, his email address and a hash of his password. This information is stored in the legacy database.

Key Pairs Consist of a private key and the according public key that the CA-Core can generate on request. They are stored permanently in the archive. It is important, that the CA-Core destroys his record of the private key as soon as possible.

Certificates A certificate that is signed by the CA-Core. It is also stored in the archive and additionally also in the CA-Storage (to allow certificates to be revoked).

Certificate Revocation List A list of certificates, that have been revoked by the CA-Core.

Local Users Credentials, that are used for system administration and communication between the components of the system.

Configuration Files Configuration files are binary files that configure configurable services like configuration configuring configurations.

Log Files Log files. Stored on the backup server.

1.4 Interfaces

Not yet standardized.

1.4.1 Web Interface

1.4.2 Legacy DB Interface

1.4.3 CA-Core Interface

1.4.4 Archive Interface

2 Risk Analysis and Security Measures

2.1 Information Assets

2.1.1 Physical Assets

Each of the machines mentioned in the system description would be considered a physical asset and would have to be protected against violation of their physical integrity. Since we are working in a virtualized lab environment, we ignore the physical assets.

2.1.2 Logical Assets

Software Includes operating systems and applications running on the physical assets. All software assets are configured and regularly updated by the system administrator.

Main Firewall Running IPCop 2.0.6. Provides Firewall functionality, routing, NAT and VPN.

Web Server Running . . . linux and . . . web server.

Second Firewall Application level Firewall with like Zorp 3.4.6.

Legacy DB Server Running . . . linux and MySQL 5.6.14.

CA-Core Server Running . . . linux with OpenSSL 1.0.1e.

Archive Firewall Running IPCop 2.0.6. Provides Firewall functionality.

Archive Server Running . . . linux.

Backup Server Running . . . linux.

Information/Data User Information The value of username and password depends on the role of the user. Since all users can create and revoke certificates the information should be protected form unauthorized access. The same is true for changing the user information. User information of a CA admin enable slightly more functionality, but these are only for displaying information not changing. The state space of this asset corresponds to the set of people who have access to the

valid username/password combination. For displaying and changing authorization has to be given, for the password confidentiality is necessary.

Key Pairs A pair of private and public Key. The state space of this asset corresponds to the set of people who should have access both public and private key which is only the holder of the private key (The keys in the archive should not be accessible in general). Confidentiality of the private key has to be guaranteed all the time.

Certificates The certificate asset consists of the certificate and the person in the DB associated with the certificate. The state space of this asset corresponds to the set of people who should have access to the certificate, which should be everybody. The relation between the user information and the certificate is fixed, if one of both changes, the information becomes invalid.

Certificate Revocation List A list of certificates, that have been revoked, availability of this List is essential.

Local User Information Credentials, that are used for system administration and communication between the components of the system. This information is different for every system. The state space of this asset corresponds to the set of people who have access to the valid username/password combination which should only be one, the system administrator. Confidentiality of both username and password has to be ensured.

Configuration Files Describe the configuratons running on the software assets. Confidentiality of these files should be ensured.

Log Files Hold information about what happened on all the systems. For auditability the logs should not be writable by the system administrator and confidential for other entities.

2.1.3 Persons

At this point we list all the personnel that are involved in creating and revoking certificates as well as persons with access to information that gets produces in these processes.

User Can create and revoke certificates and change user information. Is responsible for the confidentiality of his password and private keys on his own machine.

CA Administrator Same as for user only with a bit more read access.

System Administrator Maintains the server infrastructure. Obviously the system administrator has access to all critical data on the system. Should not have access to the private keys on the CA-Core as well as in the archive.

2.2 Threat Sources

In this Lab we are using a virtual environment and therefore neglect the environmental thread sources.

Users: Everybody who has an entry in the Database and crates and revokes certificates.

Employees: Which is namely just the system administrator and the CA administrator.

Script Kiddies: Since the systems considered are connected to the Internet, they are exposed to attacks by script kiddies.

Skilled Hacker: Skilled hackers are one of the biggest concerns to the system. They could try ton issue certificates in the name of the CA authority or gain access to private keys.

Malware: Of course malware must be taken into account. Although it is unlikely that directed malware will be used to attack the CA authority , there is still the problem of undirected malware.

2.3 Risks and Countermeasures

List all potential threats and the corresponding countermeasures. Estimate the risk based on the information about the threat, the threat sources and the corresponding countermeasure. For this purpose, use the following three tables.

Impact	
Impact	Description
High	The event may result in loss of confidentiality of private keys or root access to the CA-Core which enables the issuing of certificates.
Medium	The event may result in loss of availability of the service or loss of confidentiality of the user data but not private keys and root access on CA-Core.
Low	The event may result in an unwillingly revocation of certificates or temporarily unavailability of the service.

Likelihood	
Likelihood	Description
High	The threat source is highly motivated and sufficiently capable of exploiting a given vulnerability in order to change the asset's state. The controls to prevent the vulnerability from being exploited are ineffective.
Medium	The threat source is motivated and capable of exploiting a given vulnerability in order to change the asset's state, but controls are in place that may impede a successful exploit of the vulnerability.
Low	The threat source lacks motivation or capabilities to exploit a given vulnerability in order to change the asset's state. Another possibility that results in a low likelihood is the case where controls are in place that prevent (or at least significantly impede) the vulnerability from being exercised.

Risk Level			
Likelihood	**Impact**		
	Low	Medium	High
High	Low	Medium	High
Medium	Low	Medium	Medium
Low	Low	Low	Low

2.3.1 *Evaluation Legacy DB & User Information*

No.	Threat	Implemented/planned countermeasure(s)	L	I	Risk
1	Alter information in the DB or DB itself without authorized access to do so	Prevent code injection in DB. Only open ports to webserver & database that are needed for operation (firewall).	*Medium*	*Medium*	*Medium*
2	Read out or get access to user information and/or passwords	Prevent access to DB server. Encrypt passwords.	*Medium*	*High*	*Medium*

2.3.2 *Evaluation Certification Revocation List*

No.	Threat	Implemented/planned countermeasure(s)	L	I	Risk
3	Revoke certifications without holding the certificate	Ensure authorization before actions.	*Low*	*Low*	*Low*

2.3.3 *Evaluation CA-Core Server & Local User Information*

No.	Threat	Implemented/planned countermeasure(s)	L	I	Risk
4	Issue certificates in the name of the CA	Limit access to machines via VPN and firewall & ensure authorization before to get root privileges	*Medium*	*High*	*Medium*

2.3.4 *Evaluation Key Pairs a & Archive*

No.	Threat	Implemented/planned countermeasure(s)	L	I	Risk
5	Access to private key during transmission of certificate to user after issuing it	Encrypt communication between CA-Core and webserver as well as webserver and user	*Medium*	*High*	*Medium*
6	Access to private key by accessing the CA-Core	Instantly delete private keys on CA-Core after issuing to user and archiving. Protect access to server	*Low*	*High*	*Low*
7	Access private key by accessing and decrypting backup stored in archive	Protect access to archive, encrypt key backup and protect key for encryption	*Low*	*High*	*Low*

2.3.5 *Evaluation Local User Information & all Servers*

No.	Threat	Implemented/planned countermeasure(s)	L	I	Risk
8	Root access/login on every hardware other then the web server	Only allow access via SSH from within network (VPN), use different and strong authentication on every server	*Medium*	*High*	*Medium*

2.3.6 *Evaluation all hardware & software*

No.	Threat	Implemented/planned countermeasure(s)	L	I	Risk
9	Denial of service	Not much, no confidentiality of key parts are at stake. Could try load balancing & getting additional hardware/bandwidth	*Medium*	*Low*	*Low*

2.3.7 Risk Acceptance

No. of threat	Proposed countermeasure including expected impact
1	Allow only users to alter data when authenticated and operating on an encrypted connection. Don't allow altering the DB, just update the fields belonging to the user
2	Enforce use of strong passwords
4	Enforce strong passwords, implement two factor authentication for access to critical hardware/software. Don't let processes run with root privileges if not necessary
5	In addition to secure channel, decrypt public and private key on transmission
6	Never write key to disk, encrypt and send to user and archive
7	Enforce strong passwords, implement two factor authentication for access to critical hardware/software.
8	Enforce strong passwords, implement two factor authentication for access to critical hardware/software