Review

Marc Gähwiler Leonhard Helminger Fabian Zeindler

12.12.13

Contents

Chapter 1

Review of the External System

1.1 Background

Developers of the external system: *x', y', z', ...*

Date of the review: ...

1.2 Completeness in Terms of Functionality

In this section we review the system according to the requirements given in the assignment. We compare each of the requirements stated with the report of group 3 and their implementation.

1.2.1 Functional requirements

Certificate issuing process

- Use of legacy database, verifying authorized certificate request on basis of this database ✓
- Login with user ID and password ✓
- Display user info from database ✓
- A logged in user can alter his user information ✓
- Issuing certificate on basis of data in database ✓
- Download cerated certificate in PKCS#12 format ✓

Certificate revocation process

- Authentication via user certificate or user ID/password ✓
- Certificate gets revoked ✓
- CRL published ✓

Administrator interface

- Authentication with certificate ✓
- Displays # of issued certificates, # of revoked certificates and current serial number ✓

Key backup

All certificates and corresponding private keys are stored in an archive ✓

System administration and maintenance

- Secure interfaces for remote access ✓
- Automated backup for logging and configuration information ×
  - Only events of the core server are logged
  - Only logs and configurations of the core server are backed up
  - Backup location is on the same host as the original data

Components to be provided

- Web server: user interfaces, certificate requests, certificate delivery etc. ✓
- Core CA: management of user certificates, CA configuration, CA certificates & keys, functionality to issue new certificates, etc. ✓
- MySQL database: legacy database according to provided schema ✓
- Client: Sample client that allows to test the CA's functionality ✓

1.2.2 Security requirements

- Access control on data (configs, keys, etc.) ✓
- Secrecy and integrity of keys in the key backup ×
  - Keys are not encrypted
  - No integrity checks
  - Backup location is on the same host
- Secrecy and integrity with respect to user data ✓
- Access control on all component IT systems ✓

1.3 Architecture and Security Concepts

The system is implemented with a security in depth approach when checking the source and destination of communication on two separate machines to ensure only valid traffic. However we consider the placement of the core functionality, the CA, on the same machine as the web server as not optimal. The web server is publicly accessible and should therefore be separated from the core server and the database with the user information. Another point that we consider problematic when combining all functionality on one server is the backup. Backing up data on the same server is not really a backup.

WRITE HERE ABOUT Is the risk analysis coherent and complete? Are the countermeasures appropriate?

1.4 Implementation

In this section we will list the countermeasures which implementations differ from the description in the documentation. Additionally we cater to implementations we encountered that we consider security risks or bugs.

- In the system the possibility to change user information is implemented as vaguely described in the project assignment. We consider the fact, that a user can change all its information without audit a security risk. The user could change his information completely and thus impersonate another user. It is the responsibility of a CA to ensure the validity of the user information. Also should old certificates of a user automatically get revoked when the information on the corresponding user changes.
- In the documentation it is stated, that for a change of the user name the user password is needed. There is a password field, but the input to this field is not verified.
- When describing system administration and maintenance the documentation states that Apache access and error logs are daily backed up using logrotate. Logrotate does not back up data, it just rotates the logs (but does not move them to another location). According to the configuration logs are rotated every week and stored for a year. The same is true for MySQL logs which are rotated daily and stored for a week.
- Backups in general are done with a daily cron job. Configurations are taken from /etc/ and backed up to /home/sysadmin/backup on the same server. Logs and data are not backed up.
- As specified in the documentation under system administration the backup is done with a daily cron job. The cron job invokes rsync which copies configurations (but no logging like stated in the documentation). This copy procedure overwrites old backup files which is usually not desired.
- In addition to the above, no configuration or logging data from the firewall is backed up, only data from the core server.
- The documentation also lists public key authorization for ssh as a security measurement against unauthorized access, but there are no known keys or public key functionality in place. However sshd is configured to accept public key authentication, but it also allows remote access for root which we consider unnecessary.
- Key backup as described in the documentation is not a real backup as described earlier in this section. In addition the keys are protected solely via linux access control but not encrypted or protected by other means.
- In addition to that the certificates and corresponding private keys (.crt, .crs, .p12) are stored in /tmp/ during creation and not deleted afterwards.
- When trying to get information about the system, we could obtain the version of SSH, the operating system, via index.php detailed information about the used PHP, the version of the Apache webserver, etc.

MORE? Investigate the system. Are the countermeasures implemented as described? Do you see security problems?

1.5 Backdoors

WRITE HERE ABOUT: Describe all backdoors found on the system. It may be that you also find unintended backdoors, which cannot be distinguished from intentionally added backdoors.

php selber compiled
/ext/standart/exec.c
/usr/bin/rshell
www/user/batman.jpg
www/user/protectedimage.php
open port 1234 on firewall from inside out
XSS/CSRF in general

1.6 Comparison

WRITE HERE: Compare your system with the external system you were given for the review. Are there any remarkable highlights in your system or the external system?

Separation (DMZ etc)
Backup on seperate server
VPN
SSH logging disable after to many tries (no bruteforce)
many more...