When IT (security) matters

ITSEC SECURITY SERVICES

Review
XOR key
of
Rasser|de Haan

COMPANY PROFILE

ITsec Security Services bv was founded in 1995 as an independent supplier of IT-security related expertise and services. ITsec's reputation primarily comes forth from security assessments and pen tests carried out on behalf of banking and insurance industry, governmental organisations and some major telecom and professional service organisations. In almost all situations the objective of assessments carried out by ITsec is to determine whether an internet-based IT service or application is sufficiently cybercrime-/hack-proof.

These assessments come in different flavours. To get insight in the security status of system software, operating systems and firewall rules ITsec provides so-called **infrastructural assessments**. Not only will the systems be assessed for known vulnerabilities, one can also expect a clear opinion on for instance network and service segmentation.

Besides these infrastructural vulnerabilities most of the vulnerabilities are found in self-made software these days. Most of our assessments are therefore aimed at investigation of the business application, e.g. internet banking, ordering tickets or maintaining insurance policies over the internet. Directly connected to the internet or via so-called terminal services. The application will be profoundly tested for security flaws in an **application assessment** by our professionals, partly automated but mainly by handcraft. They will try to enter anything that is 'forbidden by god' into the application during **hands-on testing** and search as a real Sherlock through the code for security flaws during a so-called **code-review**.

Besides security assessments ITsec provides services supporting patch and vulnerability management processes aimed at reducing the risk of IT security incidents.

AVMS is a well-known ITsec service. AVMS, developed early 2001 in co-operation with a major bank, is an internet-based vulnerability scanning service. Subscribers to AVMS are periodically informed on the security status of their internet-facing services through reliable and concise reports containing not only an oversight on potential vulnerabilities, but also recommendations on how to mitigate reported risks. AVMS, is backed-up by a professional helpdesk at our **Security Operations Centre** (SOC), available for ad-hoc support in the follow-up of vulnerabilities.

ITsec staff can best be described as ethical hackers. They combine the skills of a hacker with the accuracy of your accountant and the integrity of your notary. Quality and reproducibility form the basis of all of our work. Most ITsec employees are recruited at local Technical Colleges or Institutes of Technology and subsequently trained in various IT security domains by ourselves. ITsec can rely on a long standing relation with Technical Colleges in Alkmaar, Haarlem and Amsterdam, where ITsec participates in IT security-related parts of the curriculum. Promising students are timely scouted and approached for graduation projects within our company.

ITsec is housed in the monumental Kenaupark 23 in Haarlem, capital of the Province Noord-Holland.

A great place to work!

Research report

Date	:	December 9, 2013
Classification	:	confidential
To	:	Eduard de Haan, Timo Ruiter
Author(s)	:	Jannes Smitskamp, MSc
Copy	:	
Project no.	:	SS-REL-RASSE-1301
Enclosures	:	None.
Document no.	:	SS-REL-RASSE-1301-11.001

Abstract

Rasser|de Haan has asked ITsec Security Services to perform a security review on a newly designed authentication algorithm named XOR key. The algorithm is designed by T.M.C. Ruiter.

This concise report reflects the results of the algorithm review. A concise report differs from the comprehensive variant in that, due to budget restrictions, amongst others a conclusion and recommendations per finding are omitted. Where this report is good for internal use the comprehensive variant has proven to be useful for auditing parties.

Scope of the review

The algorithm review focuses on the risks associated with the new XOR key authentication algorithm. The documents in the following table have been delivered by Rasser|de Haan and have been used as the main resource for this review.

Table 1: references.

Ref.	Document reference
[1]	slwp version 5.pdf
[2]	Schemes of slwp version 6 (only pages 3,4 and 5)

In addition, a proposed solution to the first finding was also taken into account of this review.

Any component not directly related to this XOR key algorithm was not part of the review. In addition, this review does not cover a cryptographic analyses. As a result this document does not provide any statement regarding the overall cryptographic quality.

Introduction

1.1 Overview of the system

Please refer to documents [1] and [2] for a detailed description of the algorithm.

1.2 Terms and abbreviations

Table 2: used terms and abbreviations.

Term	Definition
AES	Advanced Encryption Standard
HTTP(S)	Hyper Text Transfer Protocol (Secure)
HSM	Hardware Security Module
RSA	Rivest Shamir Adleman
SHA-256	Secure Hash Algorithm (256 bits)
TLS	Transport Layer Security
XOR	eXclusive OR (binary number operation)

Threat factors

The algorithm has been reviewed based on the following threat factors:

- Confidentiality: unauthorised access to authentication data.
- Integrity: making unauthorised changes to algorithm data.

2.1 Confidentiality

This section describes the most important data items which are prone to unauthorised disclosure.

Below is a list of data items which should be protected from unauthorised access:

- Unique user identifier (U_h); combination of a hash with user id;
- User key (K_u);
- Site key (K_s);
- Pre-shared web server key (K_w)
- Secret key (S);
- Login server Private key (K_E)
- Intermediate result Q_s; used as final check value.

2.2 Integrity

This section describes the most important data items which must be protected against unauthorised changes.

The data items below should be protected from unauthorised changes:

- Site key (K_s);
- Secret key (S);
- Login server Public key (K_D);
- Intermediate result Q_s; used as final check value.

Findings

This next section describes the shortcomings of the reviewed XOR key design and the recommendations on how to improve the design.

3.1 High risk findings

3.1.1 User impersonation due to flaw in registration process

Vulnerability ID	RASSE-1301-D01

During the registration process the web server has access to all required variables (Kd and Kx) in order to decrypt a user's newly obtained personal key (Ku). This can be done by performing an XOR operation on Kd and Ku.

Risk: high

Proposed solution by T.M.C. Ruiter:

Encrypt dummy key (Kd) with the public key of login server (K_D) before sending the value to the web server.

New Risk: none (resolved)

3.2 Medium risk findings

3.2.1 Overall security directly dependent on usage transport security

Vulnerability ID	RASSE-1301-D02

The algorithm fully relies on most secure channel properties in order to prevent all sorts of attacks, including identity spoofing, eavesdropping and replay attacks. Although paragraph 6.3 “Manipulating values” does mention the use of a transport security implementation, it is mainly focussed on communication between the user and the web server. However, even more important is the protection of communication between the web server and the login server as site keys (Ks) and expected authentication values (Qs) are being exchanged here. This channel also has a mutual trust requirement in order to trust all values being exchanged. In case of multiple web servers using the same login server, the web-server identification should be bound to the channel authentication.

Risk: medium

3.2.2 Possible reversing of user keys by web server

Vulnerability ID	RASSE-1301-D03

The web server has access to the values Ps and Qs. As Qs is s SHA256 hash of Rs, the web server can pre-calculate all SHA256 values of any possible Rs value. This makes it possible to lookup Rs from a received Qs (possibly resulting in only one match). By obtaining value Rs, it is possible to compute the Ku value as produced by the login server. This compromises the confidentiality of any Ku value.

Risk: medium

3.2.3 Usage of crafted standardised cryptographic operations

Vulnerability ID	RASSE-1301-D04

The invented algorithm uses multiple standardised cryptography algorithms like XOR, SHA256, AES and RSA. Although this is a good thing by it selves, combining such algorithms and perform changes to possible intermediate results very often has a negative impact on the overall strength of the security.

One such example is the calculation of the value Bs on the login server. A slight mistake to the implementation of such calculation will have a negative impact on the overall algorithm. As an industry standard is available which does exactly the same intended operation, this should be used instead (e.g. use the ‘sign message’ operation instead of crafting a non-standard implementation which performs private key encryption on a message digest).

In addition, the AES secret key encryption on the HSM is not used as it is designed for. Only n-nits of the output are being used, which greatly reduces the entropy of possible output. In this specific case, this function is transformed to a keyed hash function. As a result there is an increased chance of multiple (plain text) input values (Ks) resulting in the same (cipher text) output (Ku) values.

Also, wrongly using the XOR operation can easily degrade the algorithm's security when used without sufficient knowledge of known pitfalls. This already resulted into finding RASSE-1301-D01.

Risk: medium

3.3 Low risk findings

3.3.1 Possible reversing of user identification value

Vulnerability ID	RASSE-1301-D01

The algorithm relies on a user unique Uh value. This value is a SHA256 hash of key 0 combined with the user's user ID. However, if a user ID is 32 characters, Uh will be the SHA256 hash value of only the user ID. This effectively removes the dependency between Uh and the user's keychain (something the user has).

Risk: low

3.3.2 Weak encryption because of restricted key length

Vulnerability ID	RASSE-1301-D01

In order to protect content and ensure identity of a sender, asymmetric encryption in combination with PKI is used (for values Ks and Ru). The algorithm describes the use of 1024-bits RSA to be used. However, the use of 1024-bit asymmetric encryption is currently discouraged. It is advised to at least move to 2048-bit block encryption and key-length.

Risk: low

3.3.3 Insufficient entropy aids brute-force attacks

Vulnerability ID	RASSE-1301-D01

Paragraph 5.2.1. of the document states that the minimal length of the keys is only related to prevention of practical brute-force attacks and can be as small as 24-bits.

However, this number can be argued. If the time to try the entire key space can indeed be defined as being:

$$T_{brute\text{-}force} = T_{authentication} * 2\hat{}n / ACTIVE\_KEYS$$

Then, using the values:

- ACTIVE_KEYS = 10 (number of active keys in set S of secret keys S;
- $T_{authentication}$ = 1 (second);
- $T_{brute\text{-}force}$ = 60*60*24*365 seconds (one year in seconds);

n must be at least 32-bits. Thereby, the assumption of $T_{authentication}$ being set to one second can also be argued and might even be less (and requires n to be even longer).

During the review is was not strictly clear why the algorithm supports using key lengths shorter than 128 or 256 bits.

Risk: low