SafeCom is a printing solution designed for big companies. It offers print managment, security and cost reduction.

Description
Having sent a print job to a central server, the End User can pick it up on any printer or multifunction peripheral device (MFP). For this purpose, users identify themselves in the system by a proximity card and then follow the instructions displayed on the device's screen.
Various components of the system (including the print server, multifunction devices, and the administration interface) communicate with each other via an internal, closed SafeCom protocol. These connections are implemented among others on TCP ports: 7500 and 7700.

SafeCom was recently (2012) acquired by Nuance Communications.

Vulnerabilities, by the time of finding, were present in the current versions of software:
Product URL: http://www.safecom.eu/Solutions/Benefits/Secure-print.aspx
Product Name: SafeCom Pull Print™
Client version: SafeCom Print Client G4 (v. S82 070.510*03)
Server version: SafeCom G4 Server (v. S82 070.510*01)

Credits:
Jakub Kaluzny, SecuRing
together with
Sławomir Jasek, SecuRing

Dates:
Vendor contact (e-mail & phone) – 2013-10-10
Vendor response – 2013-11-18 – transferred to roadmap alignment
Public disclosure – 2014-05-21 (Positive Hack Days conference, Moscow)

1. Improper use of cryptographic mechanisms in the internal SafeCom protocol (CVE-2013-6290)

Summary
SafeCom internal protocol used for communication between various elements of the system (the central print server, the printer, and the administration application) incorrectly implements cryptographic mechanisms used for party encryption and authentication. As a result, any transmission party can be impersonated by an attacker, and thus - eavesdropping of active transmission ("Man in the Middle" attack against an encrypted protocol) or unauthorized connections to the system (such as a printer to a server) are possible. In addition, due to improper use of the symmetric block encryption algorithm, statistical analysis of the cipher text is possible, which may reveal the key or plaintext of sent messages.

Details
The steps for SafeCom protocol encrypted communication are as follows:

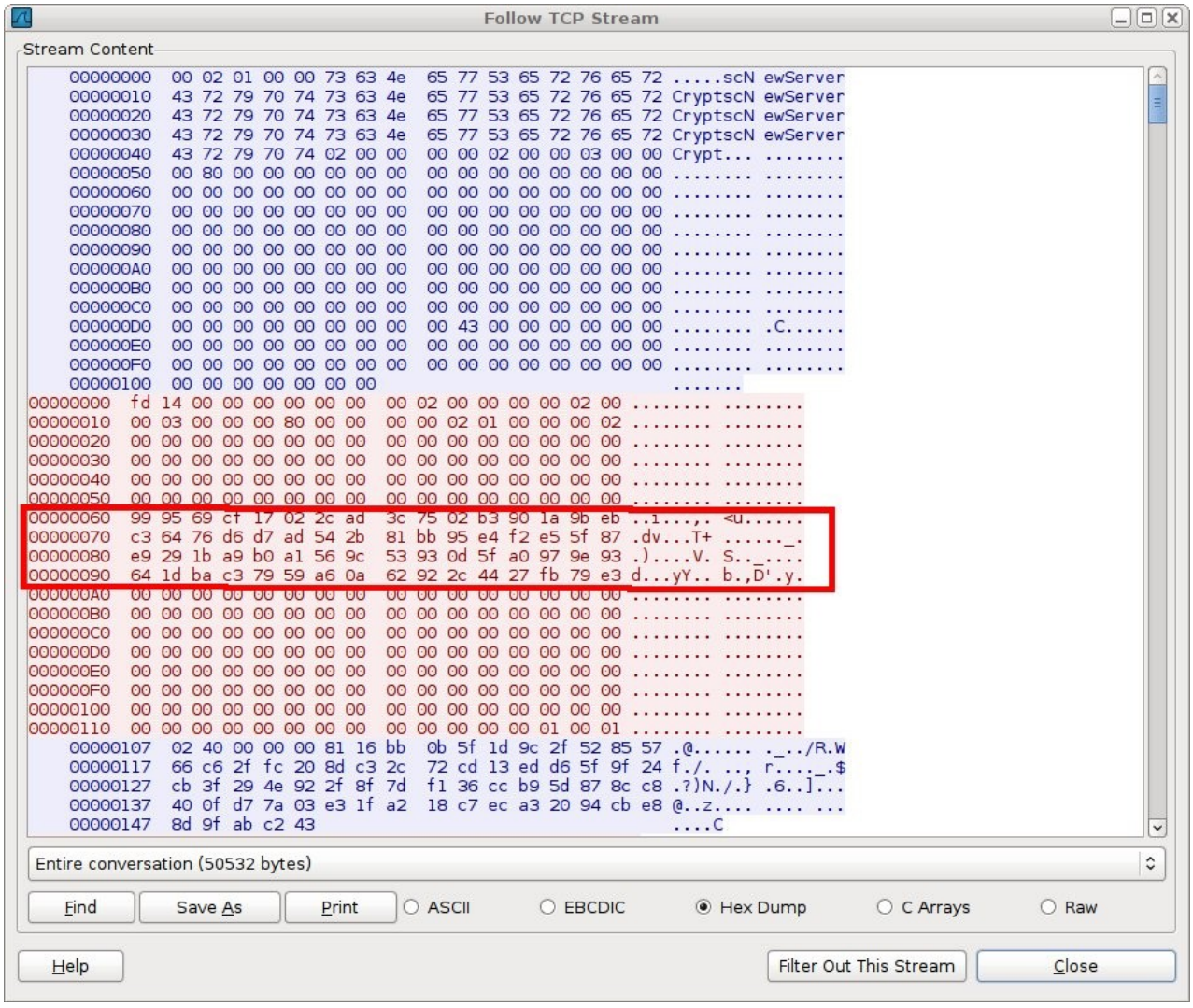
1. Client (e.g. a printer) sends its public key to a server (an RSA algorithm with key length of 512, 1024 or 2048 bits). This key is not constant - it changes, for example, when the printer is activated/restarted.
2. The server establishes a temporary, shared session key (for use in further transmission stage).
3. The server encrypts a session key with the use of the public key obtained in step 1, and sends it to the client.
4. The client decrypts the message received from the server using the secret private

key (which is a pair of the public key sent to the server in step1).

5. Thus established a secret session key is used to encrypt further transmission using symmetric algorithm (AES-128, AES-256, or Twofish-128).

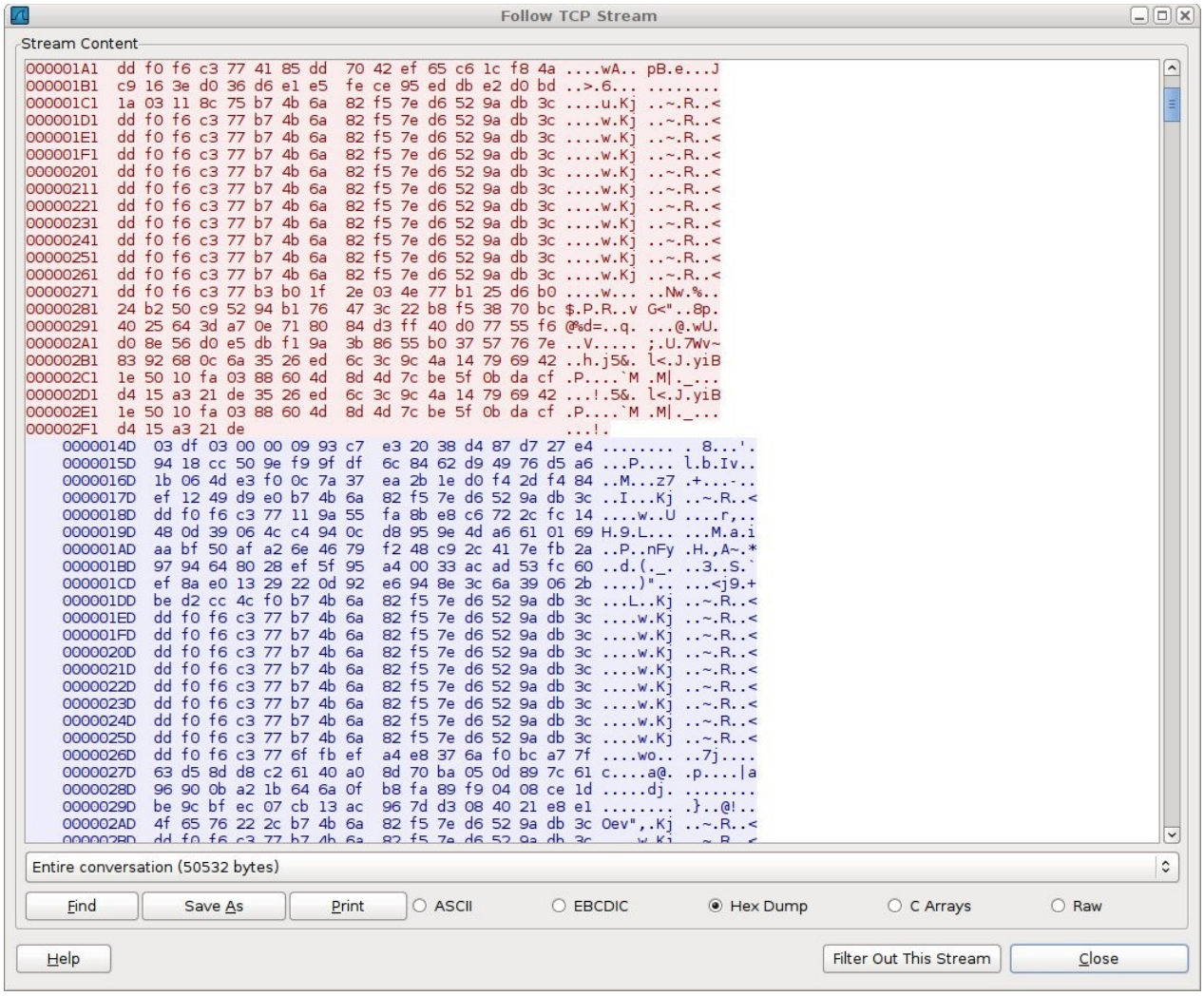
The tested solution does not apply a mechanism for verifying a client public key sent in step 1. A client simply cannot verify the server key; because such a step was not anticipated (the server does not send any data allowing its authentication). As a result, the test protocol is vulnerable to "Man in the Middle" attacks. Such attack can result in the eavesdropping and modification of messages between two parties without their knowledge.

In the tested solution, the client public key is sent at the beginning of transmission. This is part of a package indicated in the screenshot below:



It is enough to change the indicated value into own key to decipher the session key sent by the server in the next step, which enables to take complete control of transmission.

In addition, in the encrypted transmission excerpt attached below, the repetitive 16 bytes long (128 bits) blocks of data are clearly visible, indicating the use of a block cipher algorithm in ECB mode with a key length of 128 bits:



In the ECB mode, the same cipher text block corresponds with the same open text block, because each block is encrypted independently. This facilitates the cipher text cryptanalysis, and in extreme cases can lead to the key (or plaintext) reveal.

Access Conditions
Access to the local network.

Impact
A direct result of this vulnerability exploitation is the ability to eavesdrop and modify the encrypted transmission. In addition, the plaintext analysis in messages sent via the protocol makes it possible to create own SafeCom protocol client. Such client could be used for example to perform unauthorized operations.

Proof of Concept
A simple PHP script to intercept, decrypt and modify TCP traffic:

Recommendation
Proper implementation of the secure session key negotiation mechanism enables also the transmission party authentication. This can be achieved e.g. by mutual review of certificates sent during transmission initialization - in respect of compliance with the signature by a trusted Certificate Authority (PKI) or with a pattern stored locally.

In symmetric algorithms, different mode of encryption should be used instead of an ECB mode (e.g. CBC).

More info
PKI: http://en.wikipedia.org/wiki/Public-key_infrastructure

Block cipher modes: http://en.wikipedia.org/wiki/Block_cipher_mode_of_operation

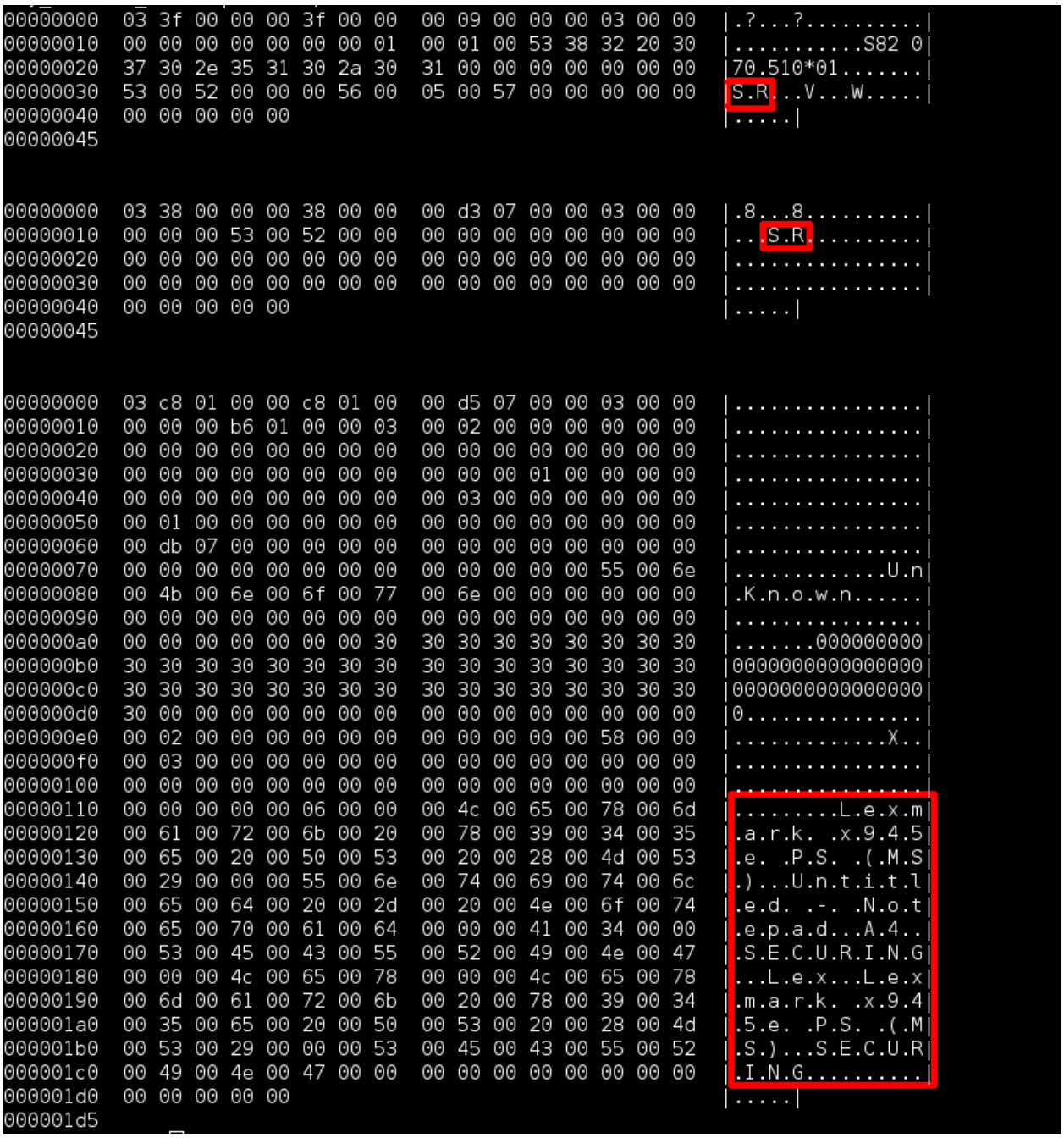
1. Ability to send print jobs on behalf of another user. (CVE-2013-6291)

Summary
Users send documents for printing through a local network to a virtual printer shared by the central SafeCom server. A printed document is located in the queue of the user currently logged on to a Windows domain. However, by simulating the SafeCom Print Client software operation, an attacker can send documents for printing on behalf of any other user.

Details
SafeCom Print Client Software is a component of the system, which can act as a local print buffer communicating with a central server via the internal SafeCom Protocol (port 7500/TCP). In this solution, the Print Client is responsible for verifying the user name sending the print job. This name is then sent in form of appropriate messages to a central server, and is not verified by the server again. So, if an attacker controlled this service, he could send a print job on behalf of any user.

During the first connection of a new Print Client instance to a central server, authorization is required by entering the username and password from the central administration server. However, during the internal SafeCom communication protocol analysis performed while testing, it was possible to pass over this requirement. A special script was prepared which simulated the operation of the SafeCom Print Client component sending a document for printing to a central server – as a test user. The document has been correctly received by the server and added to the user's queue, despite the fact that the script was run from an IP address that is not part of the pull print system, and had not previously been authenticated by an administrator.

A fragment of communication sent by the test script simulating the SafeCom Print Client operation is presented below - the first three packages with marked user name (here – „SR”) whose queue contains the document, and the document title:



Access Conditions
Access to the local network.

Impact
Sending a print job as a different user. In addition, the SafeCom Print Client has the option of direct printing (in case of no communication with the server) which probably allows printing without accountability. However, due to time constraints this test was not performed.

Proof of Concept

Recommendation
The server should not allow unauthorized connection of a SafeCom Print Client component. Furthermore, in accordance with the principle of least privilege, the system architecture should use segregated roles, and on this basis limit the opportunities available through the internal communication protocol for specific system elements.

More info

CWE-935: OWASP Top Ten 2013 Category A7 – Missing Function Level Access Control - http://cwe.mitre.org/data/definitions/935.html

2. Lack of administrator account lockout mechanism allowing to bruteforce administrative credentials (CVE-2013-6293)

Summary

This case consists of 3 low risk vulnerabilities which combined together pose a high risk on application security:

1. The ability to remotely access the interface from any point of the local network.

   Server administration is performed via a dedicated SafeCom Administrator application. Communication with the server is done by an internal SafeCom protocol at port 7700 and/or 7500 TCP. With no additional physical network restrictions, it is possible to remote log on to the system (lack of IP address whitelist).

2. Insufficient information in system logs.

   Incorrect login attempts to the administration interface are not recorded in the system logs. Only successful logins are visible in these logs, but they do not provide information on an IP address used to perform the connection.

3. The possiblity of a brute-force attack on the administrator password.

   During testing there was no limit set for incorrect login attempts, which enables brute-force attacks.

Details

Access Conditions

Access to the local network.

Impact

Obtaining administrative control over the SafeCom server. Obtaining administrative control over the SafeCom server.

Proof of Concept

20 incorrect remote login attempts were performed, followed by a successful login attempt with a correct password. The system logs did not register any information concerning incorrect login attempts, and only information on successful login was registered but without an IP address used to perform the connection.

Recommendation

Access to administrative functions should be restricted to authorized users only. Failed authentication attempts should be recorded and brute-force attempts blocked.

More info

CWE-307: Improper Restriction of Excessive Authentication Attempts - http://cwe.mitre.org/data/definitions/307.html