SIP VULNERABILITIES TESTING IN SESSION ESTABLISHMENT & USER REGISTRATION

Peter Qi Qiu, Ostap Monkewich and Robert L. Probert
School of Information Technology and Engineering
University of Ottawa, Ottawa, Canada K0J 1L0

Keywords: VoIP, signaling, authentication, vulnerability, session establishment, user registration, VOCAL, replay, spoofing, hijacking

Abstract: This paper describes an attack-directed approach to test SIP authentication vulnerabilities in session establishment and user registration. This approach aims to exercise the known areas of weakness including the inherent vulnerabilities in SIP specification and the implementation vulnerabilities caused by programmers' negligence. By using this approach and a self-made testing tool, we have successfully identified a number of vulnerabilities in a popular open source SIP implementation, namely VOCAL. This effective approach can also be used to test any other SIP implementations.

1 INTRODUCTION AND BASIC CONCEPTS

Voice over IP (VoIP), also known as packet telephony, permits the introduction of advanced user features difficult to provide over traditional telephony networks, for example, text, pictures, audio and video over the Internet. Suppliers such as Nortel Network and Cisco Systems are now selling networks with such capabilities.

SIP (Session Initiation Protocol) is the leading VoIP protocol for enabling such features. Before carrier-grade VoIP networks can supplant traditional telephony networks as revenue generators, it is essential that security issues associated with VoIP are resolved. One problem is that SIP signalling for call setup and authentication schemes are clear-text based, thus making eavesdropping by simple network sniffing a very effective type of attack. Encryption cannot be applied to all aspects of SIP messaging since SIP routers, or proxies, must understand SIP headers to perform the routing. This is one of the challenges we consider in this paper.

In fact, the Internet is the foundational service used by Voice over IP (VoIP), and the Internet itself can be considered a hostile environment from the security point of view. All Internet users must protect their transmissions from potential attackers, and SIP users are no exception. This is true for both security of SIP-enabled sessions and also for SIP signaling security. This paper will focus on authentication issues in Session Establishment and User Registration rather than on media security.

Authentication is a security feature which ensures that access is given only to users who are permitted access. During a call involving SIP user-agent and server, an attacker could masquerade as a user, forging the real identity of the sender. Authentication provides a mechanism to verify that a request sender and/or receiver are legitimate.

In a SIP-based network, the authentication can take place between the user agent and the server (proxy, registrar, and user agent server), where the server requires a user agent to authenticate itself before processing the request. Similarly, a user agent can request authentication of a server (known as Mutual Authentication).

2 CURRENT SECURITY SERVICES

Since SIP borrows its authentication mechanism from HTTP (Franks et al.,1999) (Fielding, R et al.,1999), it uses the basic and digest schemes for authentication (the basic scheme (Handley, M. et al.,1999) has been deprecated in RFC3261 (Rosenberg, J. et al., 2002)). In the basic scheme, the

Qi Qiu P., Monkewic O. and L. Probert R. (2004).
SIP VULNERABILITIES TESTING IN SESSION ESTABLISHMENT & USER REGISTRATION.
In *Proceedings of the First International Conference on E-Business and Telecommunication Networks*, pages 223-229
DOI: 10.5220/0001405402230229

client provides a user ID and password sent in clear text – a clear vulnerability. In the digest scheme, password is encrypted in MD5 (by default) algorithm (Rivest, R. et al., 1992); the server sends a response including a checksum of the nonce value, and the client returns it to the server. This scheme also represents a vulnerability since a replay attack can still succeed by simply re-sending whatever request that the attacker captures.

An important feature in aid of authentication is message integrity. Message integrity is based on an e-mail transport mechanism that transports Secure/Multipurpose Internet Mail Extensions (S/MIME) (Dusse, S. et al., 1998). Messages are signed using a public-key encryption mechanism. A SIP implementation may be tested whether or not encryption is being used regardless of the encryption quality.

Confidentiality, another related security feature, is commonly based on using encrypted-only format for messages. Signed-only and encrypted-only formats can be combined to provide authentication, message integrity and confidentiality (D. Comer, 2003).

To carry out SIP routing, end-to-end and hop-by-hop security is needed. However, proxies need to examine certain headers in order to route. Headers cannot be encrypted end-to-end, but they can be encrypted hop-by-hop. End-to-end encryption is essential because some session description protocols such as SDP used with SIP carry keys for encrypting the media. This is another area of vulnerability.

3 SUMMARY OF RELATED STUDIES

Mini-simulation (PROTOS, http://www.ee.oulu.fi-/research/ouspg/protos) is a functional method for assessing protocol implementation security. The method is designed for robustness testing that test the robustness of IUT in the face of unexpected and exceptional input. A protocol implementation that improperly handles the unexpected or malformed message may leave a security hole to some attacks. For example, a buffer overflow is possible because a programmer wrote lines of code that do not properly check the size of the destination area or buffer. A malicious user can launch a buffer overflow attack to cause the program to crash or hang.
The mini-simulation method provides a relatively simple and effective means for syntax-based robustness testing. But it is too limited to detect buffer overflow and Denial-of-Service, and it is syntax based, not checking the semantic meaning of the request/response.

In the next section, we give our approach for investigating vulnerabilities related to Session Establishment and User Registration, particularly with respect to Authentication.

4 OUR APPROACH

We use a systematic manual method for testing a SIP system for related security vulnerabilities. Our approach aims to exercise the known areas of weakness including the inherent vulnerabilities in SIP specification and the implementation vulnerabilities caused by programmers' negligence. Attack request messages are injected into the Implementation Under Test (IUT) and the system response observed. Note that our approach is not syntax-based, so the attack messages are in well form. Any response tolerating the attack request is an indicator of one or more security vulnerabilities in the IUT implementation. We have developed a test tool to inject the attack requests. By using the attack-directed test cases and test tool, we successfully identified a number of vulnerabilities in a popular open source SIP implementation, namely VOCAL (VOCAL, August 2003).

5 SECURITY THREATS AND ATTACKS

The SIP system is deployed in the Internet, a hostile environment, in which SIP elements and messages may be exposed to a variety of security threats and attacks. A threat is, by definition, a vulnerability available to a motivated and capable adversary (Bellovin, S., 1998).

This section now presents and analyses some threats that could be used to exploit the SIP implementation for the authentication aspect of security. Threats and attacks attempting to breach the lower layer encryption protection (e.g. TLS and IPSec (Dierks, T. et al., 1999) (S. Kent et al.,1998)) are not discussed in this document.

- Replay Attack

Replay attack involves a malicious user retransmitting a genuine message in order to establish authorized communication with the entity receiving the message. Replay attack is a common

threat to the client-server systems that use messages as communication means.

- Registration Hijacking

A *Registration Hijacking Attack* is a sort of replay attack conducted during the registration process. If an attacker can capture the legitimate REGISTER messages, then modify and resend a new REGISTER message to the registrar within the period set in the original timestamp, he can fool the registrar in many ways. For example, the attacker can de-register the existing registration by modifying the Expires header field with the value of 0. In this case, the original registrant (victim) cannot send or receive any calls; and cannot register his/her own device as the appropriate contact address, thereby redirecting all requests for the affected user to the attacker's device.

- Request Spoofing

Request Spoofing is used to impersonate the identity of a message sender to fool the legitimate recipient for various reasons (e.g. cheating or billing). By changing the message headers, the malicious person can send a forged request that makes the recipient believe that he/she is communicating with another entity.

Request spoofing can happen on any requests such as REGISTER and INVITE.

- Session Tear-down

This attack is to tear down a normal conversion session between the legitimate users by sending a BYE message to either user. To launch this attack, the attacker has to somehow learn the parameters (i.e. From, To, Call-ID, Cseq, etc.) of the previous call control requests, then build his own BYE message.

Vulnerability to this type of attack is caused by the lack of authentication mechanism for the BYE request in the SIP standard.

6 STRATEGY OF TEST CASE SELECTION

The high-level view of the test architecture is meant to capture the type of "attack" architecture which is common to many of the attacks experienced by networks and network services.

In order to specify the attack test cases, the test configuration shown in Figure 1 was chosen to define the relationship between the test components interconnected on a Local Area Network. The relationships of the participating components in Figure 1 were used to define a set of test scenarios and a set of test cases.

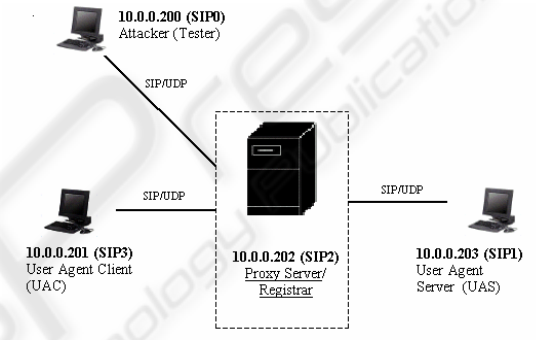

Figure 1: Test Architecture for SIP Security

The participants shown in Figure 1 are defined as follows:

Table 1: Participates in Security Testing

Computer	**Role**
SIP1	Legitimate user agent that registers with SIP2 and establishes a dialog with SIP3.
SIP2	Implementation Under Test (IUT), including Registrar and Proxy server.
SIP3	Legitimate user agent that registers with SIP2 and establishes a dialog with SIP1.
SIP0	Attacker (tester) who can capture, modify, and re-send the SIP messages exchanged among the above participants.

Our strategy of selecting test cases is attack-directed and based on the SIP specification RFC3261. This means that each test case imitates an attacker's behavior and aims to exploit the known vulnerabilities existing in RFC3261 and its implementation. The next section gives several concrete examples to illustrate our test case design and execution.

Following this strategy, we developed some attack test cases to exercise the areas of weakness in the SIP specification and implementation with the particular interest in the aspect of authentication. Note that, though these test cases by no means give an exhaustive list of the threats against SIP implementation, they represent the typical threats in authentication. These attacks include replay attack, registration hijacking, session teardown, and request spoofing.

The test cases are executed by using a self-made testing tool, namely SIP Injector. SIP Injector can send and receive SIP messages across network connections, using UDP protocol. It acts as a SIP user agent client to test other SIP entities such as SIP Registrar, Proxy, and user agent server.

In our testing, we used SIP Injector to send the IUT malicious requests such as REGISTER, INVITE, CANCEL and BYE, and to observe the response(s) from the IUT. Based on the response(s), we assign the test verdicts accordingly:

- If the IUT tolerates the malicious request, the test case is failed, meaning that the attack succeeds.
- If the IUT rejects the malicious request, the test case is passed, meaning that the attack is failed.
- Otherwise, the test case is inconclusive. In this case, the tester usually receives no responses.

7 EXAMPLE TEST CASES AND RESULTS

In this section, we use four example attacks (one for each type of attack) to illustrate how vulnerable the VOCAL registrar and proxy server are. All test cases (attacks) succeeded against VOCAL. The test cases are described in Table 2, and the test results for VOCAL is shown in Table 3. To help readers better understand the attack scenario, a message sequence diagram for the first test case (replay attack) is displayed in Figure 2.

For the first and second test cases, the results revealed the vulnerability of using a nonce value – the attacker can re-use the nonce value in the attack scenario. SIP uses a challenge/response mechanism to protect against replay attack: the server sends a nonce value to the client; the client resends the request including this nonce back to the server. Obviously, if a nonce value is only allowed to be valid in a single challenge/response transaction (also known as a one-time nonce), the replay attack will not have chance to succeed. However, the implementation of nonce value is vendor dependent, which means that vendors may choose to use the same nonce value repeatedly in the subsequent transactions in the period before the time-stamp expires. In practice, one-time nonce is rarely used due to its high cost in terms of performance. Hence, re-using nonce values opens a hole to replay attack if the attacker is able to capture and use the nonce value that has not expired yet.

Table 2: Example Test Case Description

Index	Description	Attack Type	Targeted vulnerability	Expected Result
1	An attacker records an authenticated INVITE message; then changes the Via and Contact headers to point to the attacker phone; the attacker sends the modified INVITE message to attempt to communicate with the callee.	Replay Attack	Weak nonce value in digest authentication header	Proxy rejects the INVITE message
2	An attacker records an authenticated REGISTER message; then modifies the Expires header to value 0. The attacker re-sends the REGISTER message as an attempt to de-register the legitimate registrant.	Registration Hijacking	Weak nonce value in digest authentication header	REGISTRAR rejects the REGISTER message
3	An attacker who is also a registered user, uses his user ID and password to pass the authentication, and uses another user's ID to make calls for various reasons (e.g. billing). A common form of such attack is to spoof the "From" header field.	INVITE Request Spoofing	A SIP server implementation does not compare the user ID in the From header field to the username in the Authorization or Proxy-Authorization header field.	Proxy rejects the INVITE message
4	An attacker tears down a conversation between two users by sending his own BYE request to either of the users.	Session Tear-down	SIP standard does not define authentication mechanism for BYE request.	Proxy rejects the BYE message

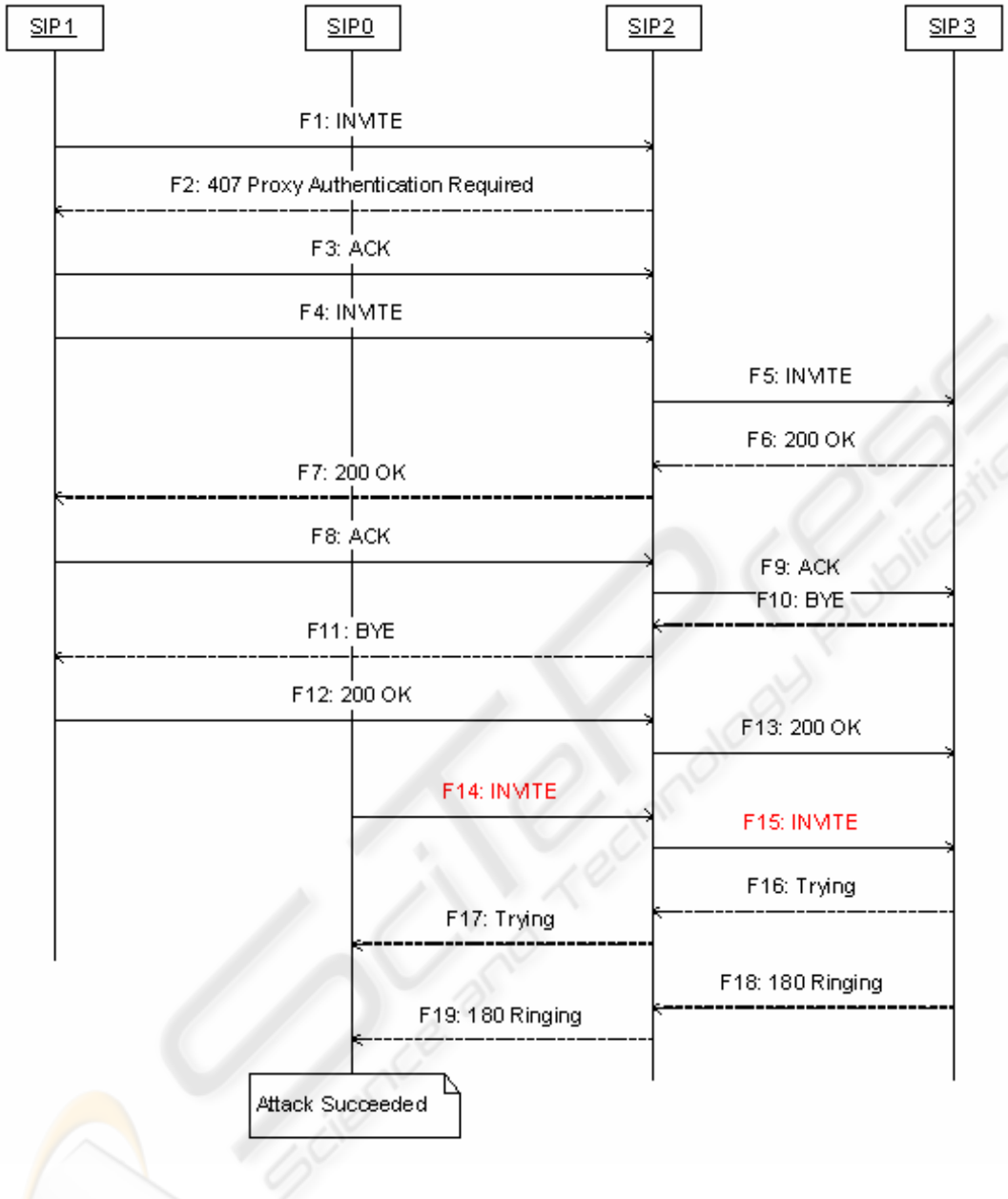

Figure 2: Message Sequence Diagram for Test Case 1 - Replay Attack

The vulnerability exploited in the third test case showed how the programmer's negligence could easily open security holes. A common form of such attack is to spoof the "From" header field in an INVITE message, or the To header field in a REGISTER message.

The fourth test case proved that a session tear-down attack is so easy to succeed due to the lack of authentication scheme for BYE request in SIP standard. The same problem exists in CANCEL request.

By following the strategy of test case selection, a tester can develop as many as test cases as desired to meet his/her test needs.

Table 3: Test Results Against VOCAL

Index	Response	Verdict
1	180 Ringing	Failed
2	200 OK	Failed
3	180 Ringing	Failed
4	200 OK	Failed

8 CONCLUSIONS

In this paper, we have presented the common threats for SIP security and an attack-directed approach to test the authentication feature in SIP implementations. More specifically, we used several examples to illustrate the vulnerabilities of authentication in VOCAL with respect to session establishment and user registration.

Although the digest authentication scheme provided with SIP can reduce the risk of attacks, there are some limitations that may open security holes to attacks. For example, replay attacks can easily succeed by simply play back the authenticated request even if the password is encrypted and a nonce value is used.

Our testing results for VOCAL demonstrated how vulnerable a SIP implementation could be if the protocol specification designers and programmers do not take precaution on security.

The next step of our testing is to execute the attack test cases on more SIP implementations. This will help us summarize the best practice and recommendations about the security services for the future SIP standard and implementation.

REFERENCES

Rosenberg, J, Schulzrinne, H, Camarillo, G, et al. *SIP: Session Initiation Protocol, RFC3261,* June 2002.

Handley, M., Schulzrinne, H., Schooler, E. and J. Rosenberg. *SIP: Session Initiation Protocol, RFC 2543*, March 1999.

Franks, et al. *HTTP Authentication: Basic and Digest Access Authentication*. RFC
2617, June 1999.

Dierks, T., Allen, C. *The TLS Protocol, Version 1.0* RFC 2246, January 1999.

Michael Thomas, SIP Security Framework, draft-thomas-sip-sec-framework-00.txt. July 12, 2001

B. Campbell, J. Rosenberg, H. Schulzrinne, C. Huitema, D. Gurle. RFC 3428 - Session Initiation Protocol (SIP) Extension for Instant Messaging, December 2002.

H. Schulzrinne, AMinimalist Security Framework for SIP, draft-schulzrinne-sip-security-00.txt. November 18, 2001.

J. Undery, S. Sen, V. Torvinen. *SIP Digest Authentication: Extensions to HTTP Digest Authentication*, draft-undery-sip-auth-00.txt. January 2002.

Bellovin, S., "Report of the IAB Security Architecture Workshop",
RFC 2316, April 1998.

Dusse, S, et al. "S/MIME Version 2 Message Specification", RFC 2311, March 1998.

S. Kent, R. Atkinson. Security Architecture for the Internet Protocol, RFC2401. November 1998.

Cisco whitepaper: Security in SIP-Based Networks. http://www.cisco.com/warp/public/cc/techno/ tyvdve/sip/prodlit/sipsc_wp.pdf. Accessed in Nov. 2003.

D. Comer, Computer and Networks with Internet Applications, Pearson Prentice Hall, 2003.

I. Dalgic, H. Fang. Comparison of H.323 and SIP for IP Telephony Signaling, http://www.nostech.co.kr/reference/data/voip/Comparison%20of%20H.323%20and%20SIP.pdf. Accessed in December 2003.

Fielding, R., Gettys, J., Mogul, J., Frysyk, H., Masinter, L., Leach, P. and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

W. Stallings. Cryptography and Network Security: Principles and Pratices, 2nd edition. Prentice-Hall, June 1998.

http://www.linuxsecurity.com/docs/Hack-FAQ/cryptology-04.shtml. Accessed in December 2003.

Rivest, R., “The MD5 Message-Digest Algorithm”, RFC 1321, April 1992.

"PROTOS - Security Testing of Protocol Implementations". University of Oulu. http://www.ee.oulu.fi/research/ouspg/protos

VOCAL: http://www.vovida.org/. accessed in August 2003.