

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 18, 2011

Shuyi Chen
ZTE Corporation
Yuting Liu
Xiaofeng Qiu
Cheng Cheng
Chunhong Zhang
MINE lab, Beijing University of Posts and Telecommunication
October 15, 2010

X.509 Extension with Security Information
draft-chen-pkix-securityinfo-00

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at
<http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on April 18, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Abstract

This document defines an X.509v3 certificate extension. It binds a list of security information to the subject of a certificate, which may be used to cognize the security posture of the subject.

Table of Contents

1. Introduction	3
1.1 Conventions used in this documents	3
2. X.509 Extension with security information	3
2.1 OID	5
2.2 Criticality	5
2.3 X.509 Security Information extension Syntax	5
2.4 X.509 Security Information extension semantics	7
3. Security Considerations	9
4. IANA Considerations	9
5. References	9
5.1 Normative References	9
5.2 Informative References	10
Appendix - ASN.1 Modules	10
Authors' Address	12

1. Introduction

This document describes an X.509v3 certificate extension that states the safety status of the certificate subject.

This certificate extension binds security information to the subject. Through this extend certificate, the subject's safety status can be obtained by the authentication entity when identity authenticating, thus to be aware of security attributes of the subject. If one entity with extend certificate with security information wants to join a certain network, network manager can evaluate entity's safety status according to its assessment standards, then make certain strategies, such as partition security level or security domain, to guarantee network safety; if the entity wants to communicate with another, it can also implements security strategy to ensure a safe between transactions, such as resource access control.

The issuer of the certificate is a trusted entity (or a trusted third party) that can identify and verify one subject's security information. Generically, security information is obtained through remote scanning measures. If can't, it is gained through local scanning by entity itself. Security threats and security protection software installed in the entity reflect the safety status of the subject directly and indirectly.

When a X.509 certificate contains an extension with security information, the extension MUST be critical, and MUST contain either a NULL to indicate that no security information is provided or explicit security information to indicate that the security information is provided.

1.1 Conventions used in this documents

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. X.509 Extension with security information

Conventional security mechanisms, such as security domain and boundary protection system, didn't take underlay safety status of the entity into consideration, which limit their range of applicability. Especially for distributed network, security protection of the nodes on underlay will influence the security degree of distributed network. Thus nodes with weak protection in underlay will greatly deteriorate security of the distributed network.

This certificate extension keeps underlay security information of the subject, and provides a basis for security strategy formulation. Based on X.509 extension certificate with security information, one entity or node can cognize another's security posture, then adjusts strategy to avoid attacks from malicious entities.

The issuer of the certificate is a trusted entity (or a trusted third party) that can identify and verify one subject's security information. Usually, security information is obtained by a trusted third party through remote scanning. Specially, if it is unable to get information through this method, it can be obtained through local scanning by the entity itself.

In general, security information is reflected in two ways. On one hand, security protection software such as Antivirus, Firewall and Operating System (OS) installed in the user reflect safety condition of the subject directly or indirectly. On the other hand, security threats such as malicious plug-ins exists in the user can also represent security status of one subject. The more threats exist, the more unsafe the entity is. Other retrievable information that can make sense for security properties of subjects can also be added according to certain needs.

Parameters of security protection software SHOULD be as specific as possible. But for private information, such as operating system software parameters, it SHOULD be abstracted as a security score, ranges in [0, 99].

The traditional X.509 certificate (without security information) has a validity period indicating the time interval during which the CA warrants that it will maintain information about the status of the certificate. Normally, it updates when the validity period is due or the key pair is no longer safe. But for certificate with extend security information, security information changes frequently. In order to ensure the accuracy of the security information in the certificate, security information MUST contain a validity period, while month or week is the unit. When this validity period is due, the certificate SHOULD be renewed. For security information updates per month/week, it increases the whole certificate update frequency. Higher update frequency increases costs.

Therefore, when certificate update is caused by security information, certificate update process SHOULD be simplified. Only update the security information of one subject without changing any other personal information that should be authenticated in certificate generation. Thus, certificate update only need to reload security information, thus there is no need of original complicated

examination process about subject personal information which is related to its identity.

An example of one use of the extend X.509 certificate with security information is a user using it to control the access of other users. Suppose both user A and B contain X.509 certificate with security information. If user A has some certain resources, and only permits access for those whose Operating System score is equal to or greater than 85. User B wants to access this resource. User A can obtain security information of user B through B's X.509 extension certificate, and determine whether it is qualified. If proved, user B can get access to the resources, else user A SHOULD refuse its access request.

X.509 extension with security information formats are as follows.

2.1 OID

The OID for this extension is id-pe-securityInfo.

```
id-pe-securityInfo OBJECT IDENTIFIER ::= { id-pe 25 }
```

where [RFC5280] defines:

```
id-pkix OBJECT IDENTIFIER ::= { iso(1) identified-organization(3)
dod(6) internet(1) security(5) mechanisms(5) pkix(7) }
```

```
id-pe OBJECT IDENTIFIER ::= { id-pkix 1 }
```

2.2 Criticality

This extension SHOULD be CRITICAL. The intended use of this extension is to indicate safety status of the identified subject. The issuer uses extended certificate to convey the notion that a relying party MUST understand the semantics of the extension to make use of the certificate for the purpose it was issued. Newly created applications that use certificates containing this extension are expected to recognize the extension.

2.3 X.509 Security Information extension Syntax

The syntax for the X.509 extension is:

```
SecurityInfo ::= CHOICE {
    none          NULL,          --No security info provided
    secInfo       SecurityInformation --Explicit security info
}

```

```

SecurityInformation ::= SEQUENCE {
    secValidityPeriod    ValidityPeriod,
    infoTime             GeneralizedTime,
    secData              SecurityData
}

ValidityPeriod ::= SEQUENCE {
    notBefore            GeneralizedTime,
    notAfter             GeneralizedTime
}

SecurityData ::= SEQUENCE {
    antivirus             (0)    AntivirusData OPTIONAL,
    firewall             (1)    FirewallData  OPTIONAL,
    operatingSystem      (2)    OSData        OPTIONAL,
    vulnerabilityDatabase (3)    VDData        OPTIONAL,
    maliciousPlug-in     (4)    MPIData       OPTIONAL,
    otherSecData         (5...MAX) ANY defined security data OPTIONAL
}

AntivirusData ::= SEQUENCE {
    antivirusBase        BasicInfo,
    otherAntivirusData   ANY defined AntivirusData OPTIONAL
}

FirewallData ::= SEQUENCE {
    firewallBase         BasicInfo,
    supFTPFileFilter     BOOLEAN,
    supAntivirus         BOOLEAN,
    supConFilter         BOOLEAN,
    defDOS               BOOLEAN,
    rtInRes              BOOLEAN,
    autoLogScan          BOOLEAN,
    otherFirewallData    ANY defined FirewallData OPTIONAL
}

BasicInfo ::= SEQUENCE {
    version              IA5String,
    manufacturer         IA5String,
    renewal              BOOLEAN
}

OSData ::= INTERGER

VDData ::= BOOLEAN

MPIData ::= SEQUENCE {
    malPlugIn            ANY defined malicious Plug-In
}

```

2.4 X.509 Security Information extension semantics

SecurityInfo is a CHOICE; it is represented either by NULL or SecurityInformation. If the issuer selects NULL, it indicates that no SecurityInfo is provided. If the issuer selects SecurityInformation, it is explicitly stating that a SecurityInfo is provided, and type SecurityInformation MUST provide details about that SecurityInfo.

SecurityInformation is a SEQUENCE consisting of three elements: secValidityPeriod, infoTime and secData. It contains all security information of one subject.

SecValidityPeriod is provided using the ValidityPeriod type. ValidityPeriod is a SEQUENCE of two GeneralizedTime values. The first (notBefore) GeneralizedTime value MUST indicate the date and time that the security information becomes valid, and the second(notAfter) GeneralizedTime value MUST indicate the date and time that the security information expires. The period of validity is in months or weeks.

InfoTime is a GeneralizedTime. It is recorded when the security information is obtained by the issuer. InfoTime type indicates when the security information is obtained exactly.

SecData is provided using the SecurityData type. SecurityData is a SEQUENCE containing security protection software and security threats. Software including antivirus, firewall and operating system are optional. Security threats MAY be reflected by vulnerabilityDatabase and maliciousPlug-in. Other software that is verified being installed in the user can also be added into this sequence. If any of software or threat elements exists in one user, its corresponding data type will be selected, then the data type in SecurityData MUST provide details of the element. If other unmentioned security data is included in the user, one can only use it after type definition.

Antivirus is provided using the AntivirusData type. AntivirusData MUST contain information about the antivirusBase and MAY contain other antivirus Data that are defined afterwards. AntivirusBase information is provided by BasicInfo type. This sequence records antivirus information, which indicates its antiviral capacity to some extent.

Firewall is provided using the FirewallData type. FirewallData is a SEQUENCE, it MUST contain firewallBase information and six boolean values, and MAY contain other Firewall Data.

Element firewallBase is also provided using BasicInfo type.

Element supFTPFileFilter is Boolean. Value one indicates this firewall support FTP (File Transfer Protocol) file filter, and allows FTP to prevent certain types of documents through this firewall; value zero is just the opposite.

Element supAntivirus is Boolean. Value one indicates this firewall support antivirus function, such as scanning the attachments of the DOC and ZIP files in E-mails to find dangerous information it may contain; value zero is just the opposite.

Element supConFilter is Boolean. If value of this element is one, it means this firewall support content filter, and MAY control the information flow according to the filter criteria. Filter content mainly refers to the URL, HTTP information--the Subject, To, From domain in Java Applet, JavaScript, ActiveX and e-mail. Value zero indicates the opposite.

Element defDOS is Boolean. Value one indicates this firewall can prevent or reduce the DOS (Denial of Service) attacks to a certain extent, while value zero is the opposite.

Element rtInRes is Boolean. It indicates whether this firewall can provide real-time intrusion prevention function. If value of this element is one, this firewall can adjust the dynamic response when invasion happens, and block malicious message. If value is zero, it indicates this firewall don't support this function.

Element autoLogScan is Boolean which indicates whether the firewall has automatic analysis and scan log function. If value is one, autoLogScan can obtain detailed log statistical results through scanning. Value zero indicates the opposite.

OtherFirewallData MAY also be added to the sequence, and can be used after definition.

BasicInfo is a SEQUENCE of two IA5Strings and a Boolean value which together specify the basis performance of the certain software. Element version contains version number information of the software. Element manufacturer use IA5String to indicate the developer of the software. The last element (renewal) MUST indicate whether the corresponding software is up-to-date. For example, an up-to-date KAPERSRY Anti-Virus V5.3 is represented as:

version	=	5.3
manufacturer	=	KAPERSRY
renewal	=	1

OperatingSystem is provided by OSData type, which is an INTEGER because OS data is private. OSData is abstracted as a security score, which indicates Operating System security status of the subject. Security score is an integer gained through local scanning of OS data information and specified calculation, ranged in [0, 99]. The bigger the numerical value is, the more safe it will be. OS data information CAN include version, manufacturer, update cycle and so on.

VulnerabilityDatabase is provided by VDDate, which is a BOOLEAN value indicates whether the vulnerability database is up-to-date. A value of zero indicates the Vulnerability Database of the subject is outdated; a value of one indicates the Vulnerability Database of the subject is up to date, which is safe.

MaliciousPlug-in is provided by MPIData. MPIData is a SEQUENCE contains any defined malicious plug-in with its details, such as name, manufacturer. The more malicious plug-in exists in the user, the less safe it is.

3. Security Considerations

This X.509 extension contains private security information, i.e., operation system information, so we abstract it into security scores to ensure confidentiality of specific information.

The trusted entity (or a trusted third party) MUST ensure that the correct values for the security information are inserted in each issued certificate, otherwise a user may reject a particular certificate if it encounters information it doesn't recognize or cannot process.

4. IANA Considerations

Certificate extensions and extended key usage values are identified by object identifiers (OIDs). The OIDs used in this document are derived from X.509 [X.509-97]. No further action by the IANA is necessary for this document or any anticipated updates.

5. References

5.1 Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

- [RFC5280] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.
- [X.690] ITU-T Recommendation X.690 (1997) | ISO/IEC 8825-1:1998, "Information Technology - ASN.1 Encoding Rules: Specification of Basic Encoding Rules(BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)".

5.2 Informative References

- [RFC 4059] Linsenbardt, D., Pontius, S., Sturgeon, A., "Internet X.509 Public Key Infrastructure Warranty Certificate Extension", RFC4059, May 2005.
- [RFC 3779] Lynn, C., Kent, S., Seo, K., "X.509 Extensions for IP Addresses and AS Identifiers", RFC 3779, June 2004.
- [X.509-97] ITU-T. Recommendation X.509: The Directory-Authentication Framework. 1997.

Appendix - ASN.1 Modules

DEFINITIONS IMPLICIT TAGS ::=

BEGIN

-- OID Arcs

id-pe OBJECT IDENTIFIER ::=

{ iso(1) identified-organization(3) dod(6) internet(1)security(5)

mechanisms(5) pkix(7) 1 }

--Security Information Extension

id-pe-securityInfo OBJECT IDENTIFIER ::= { id-pe 25 }

SecurityInfo ::= CHOICE {

none NULL, --No security info provided

secInfo SecurityInformation --Explicit security info

}

SecurityInformation ::= SEQUENCE {

secValidityPeriod ValidityPeriod,

infoTime GeneralizedTime,

secData SecurityData

}

```

ValidityPeriod ::= SEQUENCE {
    notBefore      GeneralizedTime,
    notAfter       GeneralizedTime
}

SecurityData ::= SEQUENCE {
    antivirus          (0)    AntivirusData OPTIONAL,
    firewall           (1)    FirewallData  OPTIONAL,
    operatingSystem    (2)    OSData        OPTIONAL,
    vulnerabilityDatabase (3)  VDData        OPTIONAL,
    maliciousPlug-in   (4)    MPIData       OPTIONAL,
    otherSecData       (5...MAX) ANY defined security data OPTIONAL
}

AntivirusData ::= SEQUENCE {
    antivirusBase      BasicInfo,
    otherAntivirusData ANY defined AntivirusData OPTIONAL
}

FirewallData ::= SEQUENCE {
    firewallBase      BasicInfo,
    supFTPFileFilter  BOOLEAN,
    supAntivirus      BOOLEAN,
    supConFilter       BOOLEAN,
    defDOS            BOOLEAN,
    rtInRes           BOOLEAN,
    autoLogScan       BOOLEAN,
    otherFirewallData ANY defined FirewallData OPTIONAL
}

BasicInfo ::= SEQUENCE {
    version          IA5String,
    manufacturer     IA5String,
    renewal          BOOLEAN
}

OSData ::= INTERGER

VDData ::= BOOLEAN

MPIData ::= SEQUENCE {
    malPlugIn        ANY defined malicious Plug-In
}

END

```

Authors' Addresses

Shuyi Chen
ZTE Corpoporation
17/F, ZTE Plaza, No.19, East HuaYuan Road
Haidian District, Beijing
P.R.China, 100191
Tel:+86-10-82963667
Fax:+86-10-59932043
Email:chen.shuyi@zte.com.cn

Yuting Liu
Mobile lIfe and New mEdia Lab, BUPT.
P.O. Box 92, No.10,
Xitucheng Road BeiJing, Haidian District 100876
P.R.China
Email: viviytliu@gmail.com

Xiaofeng Qiu
Mobile lIfe and New mEdia Lab, BUPT.
P.O. Box 92, No.10,
Xitucheng Road BeiJing, Haidian District 100876
P.R.China
Email: qiuxiaofeng@gmail.com

Cheng Cheng
Mobile lIfe and New mEdia Lab, BUPT.
P.O. Box 92, No.10,
Xitucheng Road BeiJing, Haidian District 100876
P.R.China
Email: chengcheng20090901@gmail.com

Chunhong Zhang
Mobile lIfe and New mEdia Lab, BUPT.
P.O. Box 92, No.10,
Xitucheng Road BeiJing, Haidian District 100876
P.R.China
Email: zhangch.bupt.001@gmail.com

Network Working Group
Internet Draft
Intended status: Informational
Expires: April 2011

J. Salowey
Cisco Systems
S. Hanna
Juniper Networks
October 18, 2010

NEA Asokan Attack Analysis
draft-salowey-nea-asokan-00.txt

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>

This Internet-Draft will expire on April 18, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Abstract

The Network Endpoint Assessment protocols are subject to a subtle forwarding attack that has become known as the NEA Asokan Attack. This document describes the attack and countermeasures that may be mounted.

Table of Contents

1. Introduction.....	2
2. NEA Asokan Attack Explained.....	2
3. Lying Endpoints.....	4
4. Countermeasures Against The NEA Asokan Attack.....	4
4.1. Identity Binding.....	4
4.2. Cryptographic Binding.....	5
4.2.1. Binding Options.....	5
4.2.1.1. Information from the TLS Tunnel.....	5
4.2.1.2. TLS Cipher Suites.....	5
4.2.1.3. Using Additional Key Material from TLS.....	6
4.2.1.4. EMA assumptions.....	6
5. Conclusions.....	6
6. IANA Considerations.....	6
7. Security Considerations.....	6
8. References.....	7
8.1. Informative References.....	7
9. Acknowledgments.....	7

1. Introduction

The Network Endpoint Assessment protocols are subject to a subtle forwarding attack that has become known as the NEA Asokan Attack. This document describes the attack and countermeasures that may be mounted.

This document is not intended to formally define a protocol but rather to explore the options for countering the Asokan attack. The NEA WG is expected to consider these options, decide which to select, and incorporate specific text defining that option into a Standards Track document. Then this document will be allowed to expire.

2. NEA Asokan Attack Explained

The NEA Asokan Attack is a variation on an attack described in a 2002 paper written by Asokan, Niemi, and Nyberg [1]. Figure 1 depicts one version of the original Asokan attack. This attack

involves tricking an authorized user into authenticating to a decoy AAA server, which forwards the authentication protocol from one tunnel to another, tricking a AAA server into believing these messages came from the attacker and granting access to him.

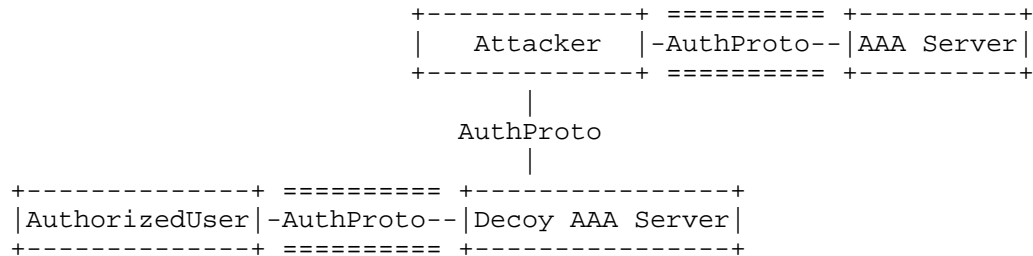


Figure 1: One Example of Original Asokan Attack

As described in the NEA Overview [2], the NEA Reference Model is composed of several nested protocols. The PA protocol is nested in the PB protocol, which is nested in the PT protocol. When used together successfully, these protocols allow a NEA Server to assess the security posture of an endpoint. The NEA Server may use this information to decide whether network access should be granted or for other purposes.

Figure 2 illustrates a NEA Asokan Attack. The attacker wants to trick GoodServer into believing that DirtyEndpoint has good security posture. This might allow the attacker to bring an infected machine onto a network and infect others, for example. To accomplish this goal, the attacker forwards PA messages from CleanEndpoint through BadServer to DirtyEndpoint, which sends them on to GoodServer. GoodServer is tricked into thinking that the PA messages came from DirtyEndpoint and therefore considers DirtyEndpoint to be clean.

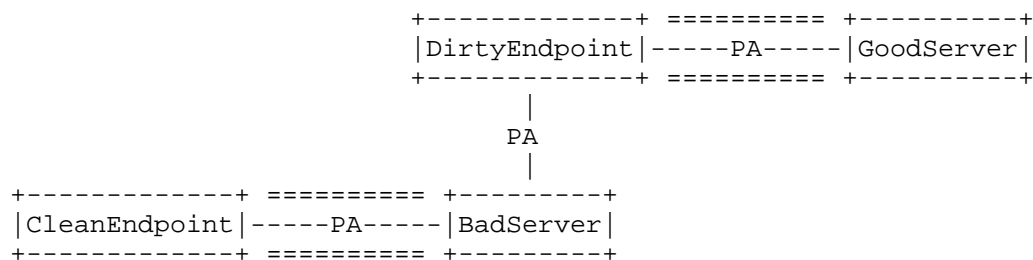


Figure 2: NEA Asokan Attack

Countermeasures against a NEA Asokan Attack are described in section 4.

3. Lying Endpoints

Some may argue that there are other attacks against NEA systems that are simpler than the Asokan attack, such as lying endpoint attacks. That is true. It's easy for an endpoint to simply lie about its posture. But there are defenses against lying endpoint attacks, such as using an external measurement agent (EMA).

An EMA is hardware, software, or firmware designed to accurately report on endpoint configuration but be especially secure and hard to compromise. The EMA observes and reports on critical aspects of endpoint posture such as which security-relevant firmware and software has been loaded.

When an EMA is used for NEA, the PA messages that reliably and securely establish endpoint posture are exchanged between the EMA itself and a Posture Validator on the NEA Server. The Posture Collector on the endpoint and any other intermediaries between the EMA and the Posture Validator on the NEA Server are not trusted. They just pass messages along as untrusted intermediaries.

To ensure that the EMA's messages are accurately conveyed to the Posture Validator even if the Posture Collector or other intermediaries have been compromised, these PA messages must provide integrity protection, replay protection, and source authentication between the EMA and the Posture Validator. Confidentiality protection is not needed, at least with respect to the software on the endpoint. But integrity protection should include protection against message deletion and session truncation. Organizations that have developed EMAs have typically developed remote attestation protocols that provide these properties. While the development of lying endpoint detection technologies is out of scope for NEA, these technologies must be supported by the NEA protocols.

4. Countermeasures Against The NEA Asokan Attack

4.1. Identity Binding

One way to mitigate the Asokan attack is to bind the identities used in tunnel establishment into a cryptographic exchange at the PA layer. While this can go a long way to preventing the attack it does not bind the exchange to a specific TLS exchange, which is desirable. In addition, there is no standard way to extract an

identity from a TLS session, which could make implementation difficult.

4.2. Cryptographic Binding

One way to thwart the NEA Asokan Attack is for the PA exchange to be cryptographically bound to the PT exchange and to any keying material or privileges granted as a result of these two exchanges. This allows the NEA Server to ensure that the PA messages pertain to the same endpoint as the party terminating the PT exchange and that no other party gains any access or advantage from this exchange.

4.2.1. Binding Options

This section discusses binding protocol solution options and provides analysis. Since the proposals for both L2 and L3 PT involve TLS the document focuses on TLS based solutions that can work with either transport.

4.2.1.1. Information from the TLS Tunnel

The TLS handshake establishes cryptographic state between the TLS client and TLS server. There are several mechanisms that can be used to export information derived from this state. The client and server independently include this information in calculations to bind the instance of the tunnel into the PA protocol.

Keying Material Export - RFC 5705 [5] defines Keying Material Exporters for TLS that allow additional secret key material to be extracted from the TLS master secret.

tls-unique Channel Binding Data - RFC 5929 [6] defines several quantities that can be extracted from the TLS session to bind the TLS session to other protocols. The tls-unique binding consists of data extracted from the TLS handshake finished message.

4.2.1.2. TLS Cipher Suites

In order to eliminate the possibility of a man-in-the-middle and thwart the Asokan attack it is important that neither TLS endpoint be in sole control of the TLS pre-master secret. Cipher suites based on key transport such as RSA cipher suites do not meet this requirement, instead Diffie-Hellman Cipher Suites, such as RSA-DHE, are required when this mechanism is employed.

4.2.1.3. Using Additional Key Material from TLS

In some cases key material is extracted from the TLS tunnel and used to derive ciphering keys used in another protocol. For example, EAP-TLS [7] uses key material extracted from TLS in lower layer ciphering. In this case the extracted keys must not be under the control of a single party so the considerations in the previous section are important.

4.2.1.4. EMA assumptions

The EMA needs to obtain the binding data from the TLS exchange and prove knowledge of the binding data in an exchange that has integrity protection, source authentication and replay protection.

5. Conclusions

The recommendations for addressing the Asokan attack are as follows:

1. Make use of cryptographic binding, however binding identities of the tunnel endpoints in the EMA may be useful.
2. The same mechanism be used in L2 and L3 PT transports that make use of TLS.
3. Neither TLS endpoint can be in sole control of the TLS pre-master secret.
4. The preferred approach is to use secret key material exported from the TLS handshake using the mechanism defined in RFC 5705. The key material is exported using a standardized label and made available to the EMA that will use it.

6. IANA Considerations

This document has no actions for IANA.

7. Security Considerations

This document is primarily concerned with analyzing and proposing countermeasures for the NEA Asokan Attack. That does not mean that it covers all the possible attacks against the NEA protocols or against the NEA Reference Model. For a broader security analysis, see the Security Considerations section of the NEA Overview [2], PA-TNC [3], and PB-TNC [4].

8. References

8.1. Informative References

- [1] N. Asokan, Valtteri Niemi, Kaisa Nyberg, "Man in the Middle Attacks in Tunneled Authentication Protocols", Nokia Research Center, Finland, Nov. 11, 2002, <http://eprint.iacr.org/2002/163.pdf>
- [2] Sangster, P., Khosravi, H., Mani, M., Narayan, K., and J. Tardo, "Network Endpoint Assessment (NEA): Overview and Requirements", RFC 5209, June 2008.
- [3] Sangster, P., and K. Narayan, "PA-TNC: A Posture Attribute (PA) Protocol Compatible with Trusted Network Connect (TNC)", RFC 5792, March 2010.
- [4] Sahita, R., Hanna, S., Hurst, R., and K. Narayan, "PB-TNC: A Posture Broker (PB) Protocol Compatible with Trusted Network Connect (TNC)", RFC 5793, March 2010.
- [5] Rescorla, E., "Keying Material Exporters for Transport Layer Security (TLS)", RFC 5705, March 2010.
- [6] Altman, J., Williams, N., and L. Zhu, "Channel Bindings for TLS", RFC 5929, July 2010.
- [7] Simon, D., Aboba, B., and R. Hurst, "The EAP-TLS Authentication Protocol", RFC 5216, March 2008.

9. Acknowledgments

The members of the NEA Asokan Design Team were critical to the development of this document: Nancy Cam-Winget, Steve Hanna, Joe Salowey, and Paul Sangster.

The authors would also like to recognize N. Asokan, Valtteri Niemi, and Kaisa Nyberg who published the original paper on this type of attack and Pasi Eronen who extended this attack to NEA protocols in the TNC.

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Addresses

Joseph Salowey
Cisco Systems, Inc.
2901 3rd. Ave
Seattle, WA 98121
USA
Email: jsalowey@cisco.com

Steve Hanna
Juniper Networks, Inc.
79 Parsons Street
Brighton, MA 02135
USA
Email: shanna@juniper.net

