Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 7, 2009

T. Clancy
LTS
K. Hoeper
NIST

November 3, 2008

Channel Binding Support for EAP Methods
draft-clancy-emu-chbind-04

Abstract

This document defines how to implement channel bindings for Extensible Authentication Protocol (EAP) methods to address the lying NAS problem.

Table of Contents

1. Introduction

The so-called "lying NAS" problem is a well-documented problem with the current Extensible Authentication Protocol (EAP) architecture [RFC3748] when used in pass-through authenticator mode. Here, a Network Access Server (NAS), or pass-through authenticator, may represent one set of information (e.g. network identity, capabilities, configuration, etc) to the backend Authentication, Authorization, and Accounting (AAA) infrastructure, while representing contrary information to EAP clients. Another possibility is that the same false information could be provided to both the EAP client and EAP server by the NAS.

A concrete example of this may be an IEEE 802.11 access point with a security association to a particular AAA server. While there may be some identity tied to that security association, there's no reason the access point needs to advertise a consistent identity to clients. In fact, it may include whatever information in its beacons (e.g. different SSID or security properties) it desires. This could lead to situations where, for example, a client joins one network that is masquerading as another.

Another current limitation of EAP is its minimal ability to perform authorization. Currently EAP servers can only make authorization decisions about network access based on information they know about peers. If the same EAP server controls access to multiple networks, it has little information about the NAS to which the peer is connecting, and does not know what information the NAS may be claiming about the network to the peer. A mechanism is needed that allows the EAP server to apply more detailed policies to authorization.

This document defines and implements EAP channel bindings to solve these two problems, using a process in which the EAP client provides information about the characteristics of the service provided by the authenticator to the AAA server protected within the EAP method. This allows the server to verify the authenticator is providing information to the peer consistent with the defined network policy, and that the peer is authorized to access the network in the manner described by the NAS. "AAA Payloads" defined in [I-D.clancy-emu-aaapay] proposes a mechanism to carry this information.

2. Terminology

In this document, several words are used to signify the requirements of the specification. These words are often capitalized. 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 [RFC2119].

3. Problem Statement

In a [RFC4017]-compliant EAP authentication, the EAP client and EAP server mutually authenticate each other, and derive keying material. However, when operating in pass-through mode, the EAP server can be far removed from the authenticator. A malicious or compromised authenticator may represent incorrect information about the network to the client in an effort to affect its operation in some way. Additionally, while an authenticator may not be compromised, other compromised elements in the network could provide false information to the authenticator that it could simply be relaying to EAP clients. Our goal is to ensure that the authenticator is providing correct information to the EAP client during the initial network discovery, selection, and authentication.

There are two different types of networks to consider: enterprise networks and service provider networks. In enterprise networks, we assume a single administrative domain, making it feasible for an EAP server to have information about all the authenticators in the network. In service provider networks, global knowledge is infeasible due to indirection via roaming. When a client is outside its home administrative domain, the goal is to ensure that the level of service received by the client is consistent with the contractual agreement between the two service providers.

The following are a couple example attacks possible by presenting false network information to clients.

- Enterprise Network: A corporate network may have multiple virtual LANs (VLANs) running throughout their campus network, and have IEEE 802.11 access points connected to each VLAN. Assume one VLAN connects users to the firewalled corporate network, while the other connects users to a public guest network. The corporate network is assumed to be free of adversarial elements, while the guest network is assumed to possibly have malicious elements. Access Points on both VLANs are serviced by the same EAP server, but broadcast different SSIDs to differentiate. A compromised access point connected to the guest network could advertise the SSID of the corporate network in an effort to lure clients to connect to a network with a false sense of security regarding their traffic. Conditions and further details of this attack can be found in the Appendix.

- Service Provider Network: An EAP-enabled mobile phone provider operating along a geo-political boundary could boost their cell towers' transmission power and advertise the network identity of the neighboring country's indigenous provider. This would cause unknowing handsets to associate with an unintended operator, and consequently be subject to high roaming fees without realizing they had roamed off their home provider's network. This scenario can be considered as "lying provider" problem, because here the provider tampers with the transmission power and then configures its NAS to broadcast another network's identity. For the purpose of channel bindings as defined in this draft, it does not matter which local entity (or entities) is "lying" in a service provider network (local NAS, local authentication server and/or local proxies), because the only information received from the visited network that is verified by channel bindings is the information the home authentication server received from the last hop in the communication chain. In other words, channel bindings enable the detection of inconsistencies in the information from a visited network, but cannot determine which entity is lying. Naturally, channel bindings for EAP methods can only verify the endpoints and, if desirable, intermediate hops need to be protected by the employed AAA protocol.

To address these problems, a mechanism is required to validate unauthenticated information advertised by EAP authenticators.

4. Channel Bindings

EAP channel bindings seek to authenticate previously unauthenticated information provided by the authenticator to the EAP peer, by allowing the client and server to compare their perception of network properties in a secure channel.

It should be noted that the definition of EAP channel bindings differs somewhat from channel bindings documented in [RFC5056], which seek to securely bind together the end points of a multi-layer protocol, allowing lower layers to protect data from higher layers. Unlike [RFC5056], EAP channel bindings do not ensure the binding different layers of a session but rather the information advertised to EAP client by an authenticator acting as pass-through device during an EAP execution.

There are two main approaches to EAP channel bindings:

- After keys have been derived during an EAP execution, the peer and server can, in an integrity-protected channel, exchange plaintext information about the network with each other, and verify

consistency and correctness.

- Network information can be uniquely encoded into an opaque blob that can be included directly into the derivation of the EAP session keys.

Both approaches are only applicable to key deriving EAP methods and both have advantages and disadvantages. Advantages of exchanging plaintext information include:

- It allows for policy-based comparisons of network properties, rather than requiring precise matches for every field, which achieves a policy-defined consistency, rather than bitwise equality. This allows network operators to define which properties are important and even verifiable in their network.

- EAP methods that support extensible, integrity-protected channels can easily include support for exchanging this network information. In contrast, direct inclusion into the key derivation would require revisions to existing EAP methods or a wrapper EAP method.

- Given it doesn't affect the key derivation, this approach facilitates debugging, incremental deployment, backward compatibility and a logging mode in which verification results are recorded but do not have an affect on the remainder of the EAP execution. The exact use of the verification results can be subject to the network policy. Additionally, consistent information canonicalization and formatting for the key derivation approach would likely cause significant deployment problems.

The following are advantages of directly including channel binding information in the key derivation:

- EAP methods not supporting extensible, integrity-protected channels could still be supported, either by revising their key derivation, revising EAP, or wrapping them in a universal method that supports channel binding.

- It can guarantee proper channel information, since subsequent communication would be impossible if differences in channel information yielded different session keys on the EAP client and server.

The scope of EAP channel bindings differs somewhat depending on the type of deployment in which they are being used. In enterprise networks, they can be used to authenticate very specific properties of the authenticator (e.g. MAC address, supported link types and

data rates, etc), while in service provider networks they can generally only authenticate broader information about a roaming partner's network (e.g. network name, roaming information, link security requirements, etc). The reason for the difference has to do with the amount of information you expect your home EAP server to know about the authenticator and/or network to which the peer is connected. In roaming cases, the home server is likely to only know information contained in their roaming agreements.

With any multi-hop AAA infrastructure, many of the specific NAS properties are obscured by the AAA proxy that's decrypting, reframing, and retransmitting the underlying AAA messages. Especially service provider networks are affected by this and the information received from the last hop may not contain much verifiable information any longer. For example, information such as the NAS IP address may not be known to the EAP server. This affects the ability of the EAP server to verify specific NAS properties. However, often verification of the MAC or IP address of the NAS is not useful for improving the overall security posture of a network. More often it is useful to make policy decisions about services being offered to peers. For example, in an IEEE 802.11 network, the EAP server may wish to ensure that clients connecting to the corporate intranet are using secure link- layer encryption, while link-layer security requirements for clients connecting to the guest network could be less stringent. These types of policy decisions can be made without knowing or being able to verify the IP address of the NAS through which the peer is connecting. Furthermore, as described in the next section, channel bindings also verify the information provided by peer and a local policy database, where both pieces of information are unaffected by the processing of intermediate hops. Consequently, even if some information got lost in transition and thus may not be known to the EAP server, the server is still able to carry out the channel binding verification.

Also, a peer's expectations of a network may also differ. In a mobile phone network, peers generally don't care what the name of the network is, as long as they can make their phone call and are charged the expected amount for the call. However, in an enterprise network a peer may be more concerned with specifics of where their network traffic is being routed.

Any deployment of channel bindings should take into consideration both what information the EAP server is likely to know, and also what type of network information the peer would want and need authenticated.

5. Channel Binding Protocol

This section defines a protocol for verifying channel binding information during an EAP authentication. The protocol uses the approach where plaintext data is exchanged, since it allows channel bindings to be used more flexibly in varied deployment models.

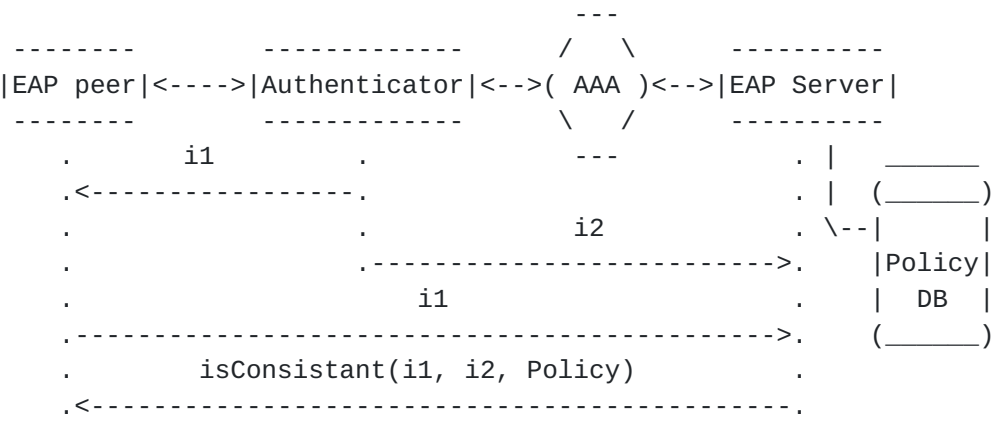

Figure 1: Overview of Channel Binding

Channel bindings are always provided between two communication endpoints, here the EAP client and server, who communicate through an authenticator in pass-trough mode. During network advertisement, selection, and authentication, the authenticator presents unauthenticated information, labeled i1 for convenience, about the network to the peer. This information, i1, could include an authenticator identifier and the identity of the network it represents, in addition to advertised network information such as offered services and roaming information. As there is no established trust relationship between the peer and authenticator, there is no way for the peer to validate this information.

Additionally, during the transaction the authenticator presents a number of information properties about itself to the AAA infrastructure which may or may not be valid. We label this information i2. Note that i2 is the information the EAP server receives from the last hop in the communication chain which is not necessarily the authenticator. In those cases i2 may be different from the original information sent by the authenticator because of en route processing or malicious modifications. As a result, in the service provider model, typically the EAP server is able to verify only the last-hop portion of i2, or values propagated by proxy servers.

Our goal is to transport i1 from the peer to the server, and allow the server to verify the consistency of i1 from the peer and i2 from

the authenticator against the information stored in its local policy database.

By doing this, we allow the EAP server the opportunity to make informed decisions about authorization. The EAP server can authenticate the authenticator via the AAA security association, and using this channel bindings mechanism it can now authorize the circumstances under which a peer connects to the authenticator.

This information, i1, could include an authenticator identifier and the identity of the network it represents, in addition to advertised network information such as offered services and roaming information. To prevent attacks by a lying NAS or lying provider, the EAP server must be able to verify that i1 either matches its knowledge of the network (enterprise model) or is consistent with the contractual agreement between itself and the roaming partner network to which the client is connected (service provider model). Additionally, it should verify that this information is consistent with i2.

The protocol defined in this document is a single round trip between the EAP peer and server that can be piggybacked to the EAP method execution, and formats data elements as Diameter AVPs. We provide requirements for a transport protocol.

6. System Requirements

The channel binding protocol defined in this document must be transported after keying material has been derived between the EAP peer and server, and before the peer would suffer adverse affects from joining an adversarial network. To satisfy this requirement, it should occur either during the EAP method execution or during the EAP lower layer's secure association protocol.

The transport protocol for carrying channel binding information MUST support end-to-end (i.e. between the EAP peer and server) message integrity protection to prevent the adversarial NAS or AAA device from manipulating the transported data. The transport protocol SHOULD provide confidentiality. The motivation for this that the channel bindings could contain private information, including peer identities, which SHOULD be protected.

If transporting data directly within an EAP method, it MUST be able to carry integrity protected data from the EAP peer to server. EAP methods SHOULD provide a mechanism to carry protected data from server to peer. EAP methods MUST export channel binding data to the AAA subsystem on the EAP server. EAP methods MUST be able to import channel binding data from the lower layer on the EAP peer.

One way to transport the single round-trip exchange is as a series of Diameter AVPs formatted and encapsulated in EAP methods per [I-D.clancy-emu-aaapay]. For each lower layer, this document defines the parameters of interest, and the appropriate Diameter AVPs for transporting them. Additionally, guidance on how to perform consistency checks on those values will be provided.

In order to minimize data formatting inconsistencies, parameters useful for channel binding MUST be allocated from the standard RADIUS space. Two AVPs are considered equivalent for the purpose of channel binding if they have the same AVP Code, Vendor-Specific Bit, AVP Length, Vendor-ID (if Vendor-Specific Bit is set), and data.

7. Lower-Layer Bindings

This section discusses AVPs of some EAP-employing lower layer link protocols that seem appropriate for providing channel bindings (i.e. data from "i1" in Section Section 5). The discussion is limited to protocols that establish fresh authentic keying material because such keying material is necessary to protect the integrity of all AVPs that are exchanged as part of the channel binding. For each protocol, a variety of network information that can be encapsulated in AVPs is of interest for server and peer to ensure channel binding. The respective appropriate AVPs depend on the lower layer protocol as well as on the network type (i.e. enterprise network or service provider network) of an application.

For each EAP lower layer, a variety of AAA properties may be of interest to the server. These values may already be known by the server, or may be transported to the server via an existing RADIUS or Diameter connection.

As part of the channel binding protocol, the EAP peer sends encapsulated AVPs to the server. The server then validates the received information by comparing it to information stored in a local database. If the received information is unsatisfactory given some validation policy, the server SHOULD respond by halting the EAP authentication and returning an EAP-Failure.

If validation is successful, the server SHOULD send a message indicating the success to the client. In addition, the server MAY respond back to the EAP peer with information encapsulated in AVPs that can be of use to the peer, and information the peer may not have any way of otherwise knowing.

7.1. General Attributes

This section lists AVPs useful to all link-layers.

NAS-Port-Type: Indicates the underlying link-layer technology used to connect (e.g. IEEE 802.11, PPP, etc), and MUST be included by the EAP client, and SHOULD be verified against the database and NAS-Port-Type received from the NAS.

Cost-Information: AVP from the Diameter Credit-Control Application [RFC4006] to the peer indicating how much peers will be billed for service and MAY be included by the EAP client and verified against roaming profiles stored in the database.

7.2. IEEE 802.11

The client SHOULD transmit to the server the following fields, encapsulated within the appropriate Diameter AVPs:

Called-Station-Id: contains BSSID and SSID and MUST be included by the EAP client, and SHOULD be verified against the database and Called-Station-Id received from the NAS

[TODO: Need a way to transport the RSN-IE.]

7.2.1. IEEE 802.11r

In addition to the AVPs for IEEE 802.11, an IEEE 802.11r client SHOULD transmit the following additional fields:

Mobility-Domain-Id: Identity of the mobility domain and MUST be included by the EAP client, and SHOULD be verified against the database and Mobility-Domain-Id received from the NAS [I-D.aboba-radext-wlan]

7.2.2. IEEE 802.11s

In addition to the AVPs for IEEE 802.11, an IEEE 802.11s client SHOULD transmit the following additional fields:

Mesh-Key-Distributor-Domain-Id: Identity of the Mesh Key Distributor Domain and MUST be included by the EAP client, and SHOULD be verified against the database and Mesh-Key-Distributor-Domain-Id received from the NAS [I-D.aboba-radext-wlan]

7.3. IEEE 802.16

TBD

7.4. Wired 802.1X

TBD

7.5. Point to Point Protocol (PPP)

TBD

7.6. Internet Key Exchange v2 (IKEv2)

TBD

7.7. 3GPP2

TBD

7.8. PANA

TBD

8. AAA-Layer Bindings

This section discusses which AAA attributes in RADIUS Accept-Request messages can and should be validated by a AAA server (i.e. data from "i2" in Section Section 5). As noted before, this data can be manipulated by AAA proxies either to enable functionality (e.g. removing realm information after messages have been proxied) or maliciously (e.g. in the case of a lying provider). As such, this data cannot always be easily validated. However as thorough of a validation as possible should be conducted in an effort to detect possible attacks.

User-Name: This value should be checked for consistency with the database and any method-specific user information. If EAP method identity protection is employed, this value typically contains a pseudonym or keyword.

NAS-IP-Address: This value is typically the IP address of the authenticator, but in a proxied connection it likely will not match the source IP address of an Access-Request. A consistency check MAY verify the subnet of the IP address was correct based on the last-hop proxy.

Called-Station-Id: This is typically the MAC address of the NAS. On an enterprise network, it MAY be validated against the MAC address is one that has been provisioned on the network.

Calling-Station-Id: This is typically the MAC address of the EAP Client, and verification of this is likely difficult, unless EAP credentials have been provisioned on a per-host basis to specific L2 addresses. It SHOULD be validated against the database in an enterprise deployment.

NAS-Identifier: This is an identifier populated by the NAS, and could be related to the MAC address, and should be validated similarly to the Called-Station-Id.

NAS-Port-Type: This specifies the underlying link technology. It SHOULD be validated against the value received from the client in the information exchange, and against a database of authorized link-layer technologies.

9. Security Considerations

9.1. Trust Model

We consider a trust model in which the peer and server trust each other. This is not unreasonable, considering they already have a trust relationship. In this trust model, client and authentication server are honest while the authenticator is maliciously sending false information to client and/or server. The following are the trust relationships:

- The server trusts that the channel binding information received from the client is the information that the client received from the authenticator.
- The client trusts the channel binding result received from the server.
- The server trusts the information contained within its local database.

In order to establish the first two trust relationships during an EAP execution, an EAP method MUST provide the following:

- mutual authentication between client and server
- derivation of keying material including a key for integrity protection of channel binding messages
- sending i1 from client to server over an integrity-protected channel

- sending the result and optionally i2 from server to client over an integrity-protected channel

9.2. Consequences of Trust Violation

If any of the trust relationships listed in Section 7.1 are violated, channel binding cannot be provided. In other words, if mutual authentication with key establishment as part of the EAP method as well as protected database access are not provided, then achieving channel binding is not feasible.

Dishonest peers can only manipulate the first message i1 of the channel binding protocol. In this scenario, a peer sends i1' to the server. If i1' is invalid, the channel binding validation will fail and the server will abort the EAP authentication. On the other hand if i1' passes the validation, either the original i1 was wrong and i1' corrected the problem or both i1 and i1' constitute valid information. All cases do not seem to be of any benefit to a peer and do no pose a security risk.

Dishonest servers can send EAP-Failure messages and abort the EAP authentication even if the received i1 is valid. However, servers can always abort any EAP session independent of whether channel binding is offered or not. On the other hand, dishonest servers can claim a successful validation even for an invalid i1. This can be seen as collaboration of authenticator and server. Channel binding can neither prevent nor detect such attacks. In general such attacks cannot be prevented by cryptographic means and should be addressed using policies making servers liable for their provided information and services.

Additional network entities (such as proxies) might be on the communication path between peer and server and may attempt to manipulate the channel binding protocol. If these entities do not possess the keying material used for integrity protection of the channel binding messages, the same threat analysis applies as for the dishonest authenticators. Hence, such entities can neither manipulate single channel binding messages nor the outcome. On the other hand, entities with access to the keying material must be treated like a server in a threat analysis. Hence such entities are able to manipulate the channel binding protocol without being detected. However, the required knowledge of keying material is unlikely since channel binding is executed before the EAP method is completed, and thus before keying material is typically transported to other entities.

9.3. Privacy Violations

While the channel binding information exchanged between EAP peer and EAP server (i.e. i1 and the optional result message) must always be integrity-protected it may not be encrypted. In the case that these messages contain identifiers of peer and/or network entities, the privacy property of the executed EAP method may be violated. Hence, in order to maintain the privacy of an EAP method, the exchanged channel binding information must be encrypted.

10. Operations and Management Considerations

This section analyzes the impact of channel bindings on existing deployments of EAP.

10.1. System Impact

As with any extension to existing protocols, there will be an impact on existing systems. Typically the goal is to develop an extension that minimizes the impact on both development and deployment of the new system, subject to the system requirements. In this section we discuss the impact on existing devices that currently utilize EAP, assuming the channel binding information is transported within the EAP method execution.

The EAP peer will need an API between the EAP lower layer and the EAP method that exposes the necessary information from the NAS to be validated to the EAP peer, which can then feed that information into the EAP methods for transport. For example, an IEEE 802.11 system would need to make available the various information elements that require validation to the EAP peer which would properly format them and pass them to the EAP method. Additionally, the EAP peer will require updated EAP methods that support transporting channel binding information. While most method documents are written modularly to allow incorporating arbitrary protected information, implementations of those methods would need to be revised to support these extensions. Driver updates are also required so methods can access the required information.

No changes to the pass-through authenticator would be required.

The EAP server would need an API between the database storing NAS information and the individual EAP server. The EAP methods need to be able to export received channel binding information to the EAP server so it can be validated.

Additionally, an interface is necessary for populating the EAP server

database with the appropriate parameters. In the enterprise case, when a NAS is provisioned, information about what it should be advertising to peers needs to be entered into the database. In the service provider case, there should be a mechanism for entering contractual information about roaming partners.

To ease operator burden it is highly recommended that there be a mechanism for automatically populating the EAP server policy database. Channel bindings could be enabled to allow peers to transmit the NAS information to the EAP server, but the policy could be configured to allow all connections. The obtained information could be used to auto-generate policy information for the database, assuming there are no adversarial elements in the network during the auto-population phase.

Channel binding validation can also be implemented incrementally. An initial database may be empty, and all channel bindings are automatically approved. Operators can then incrementally add parameters to the database regarding specific authenticators or groups of authenticators that must be validated. Additionally, a network could also self-form this database by putting the network into a "learning" mode, and could then recognize inconsistencies in the future.

10.2. Cost-Benefit Analysis

[TBD]

11. IANA Considerations

This document contains no IANA considerations.

12. References

12.1. Normative References

[I-D.aboba-radext-wlan]
Aboba, B., Malinen, J., Congdon, P., and J. Salowey, "RADIUS Attributes for IEEE 802 Networks", draft-aboba-radext-wlan-09 (work in progress), October 2008.

[I-D.ietf-dime-rfc3588bis]
Fajardo, V., Arkko, J., Loughney, J., and G. Zorn, "Diameter Base Protocol", draft-ietf-dime-rfc3588bis-13 (work in progress), November 2008.

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

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, "Extensible Authentication Protocol (EAP)", RFC 3748, June 2004.

12.2. Informative References

[I-D.clancy-emu-aaapay] Clancy, T., "EAP Method Support for Transporting AAA Payloads", Internet Draft draft-clancy-emu-aaapay-01, July 2008.

[RFC4006] Hakala, H., Mattila, L., Koskinen, J-P., Stura, M., and J. Loughney, "Diameter Credit-Control Application", RFC 4006, August 2005.

[RFC4017] Stanley, D., Walker, J., and B. Aboba, "Extensible Authentication Protocol (EAP) Method Requirements for Wireless LANs", RFC 4017, March 2005.

[RFC5056] Williams, N., "On the Use of Channel Bindings to Secure Channels", RFC 5056, November 2007.

[RFC5247] Aboba, B., Simon, D., and P. Eronen, "Extensible Authentication Protocol (EAP) Key Management Framework", RFC 5247, August 2008.

[HC07] Hoeper, K. and L. Chen, "Where EAP Security Claims Fail", ICST QShine, August 2007.

Appendix A. Attacks Prevented by Channel Bindings

In the following it is demonstrated how the presented channel bindings can prevent attacks by malicious authenticators (representing the lying NAS problem) as well as malicious visited networks (representing the lying provider problem).

A.1. Enterprise Subnetwork Masquerading

As outlined in Section 3, an enterprise network may have multiple VLANs providing different levels of security. In an attack, a malicious NAS connecting to a guest network with lesser security protection could broadcast the SSID of a subnetwork with higher protection. This could lead clients to believe that they are accessing the network over secure connections, and, e.g., transmit

confidential information that they normally would not send over a weakly protected connection. This attack works under the conditions that clients use the same set of credentials to authenticate to the different kinds of VLANs and that the VLANs support at least one common EAP method. If these conditions are not met, the EAP server would not authorize the clients to connect to the guest network, because the clients used credentials and/or an EAP method that is associated with the corporate network.

A.2. Forced Roaming

Mobile phone providers boosting their cell tower's transmission power to get more users to use their networks have occurred in the past. The increased transmission range combined with a NAS sending a false network identity lures users to connect to the network without being aware of that they are roaming.

Channel bindings would detect the bogus network identifier because the network identifier send to the authentication server in i1 will neither match information i2 nor the stored data. The verification fails because the info in i1 claims to come from the peer's home network while the home authentication server knows that the connection is through a visited network outside the home domain. In the same context, channel bindings can be utilized to provide a "home zone" feature that notifies users every time they are about to connect to a NAS outside their home domain.

A.3. Downgrading attacks

A malicious authenticator could modify the set of offered EAP methods in its Beacon to force the peer to choose from only the weakest EAP method(s) accepted by the authentication server. For instance, instead of having a choice between EAP-MD5-CHAP, EAP-FAST and some other methods, the authenticator reduces the choice for the peer to the weaker EAP-MD5-CHAP method. Assuming that weak EAP methods are supported by the authentication server, such a downgrading attack can enable the authenticator to attack the integrity and confidentiality of the remaining EAP execution and/or break the authentication and key exchange. The presented channel bindings prevent such downgrading attacks, because peers submit the offered EAP method selection that they have received in the beacon as part of i1 to the authentication server. As a result, the authentication server recognizes the modification when comparing the information to the respective information in its policy database.

A.4. Bogus Beacons in IEEE 802.11r

In IEEE 802.11r, the SSID is bound to the TSK calculations, so that the TSK needs to be consistent with the SSID advertised in an authenticator's Beacon. While this prevents outsiders from spoofing a Beacon it does not stop a "lying NAS" from sending a bogus Beacon and calculating the TSK accordingly.

By implementing channel bindings, as described in this draft, in IEEE 802.11r, the verification by the authentication server would detect the inconsistencies between the information the authenticator has sent to the peer and the information the server received from the authenticator and stores in the policy database.

A.5. Forcing false authorization in IEEE 802.11i

In IEEE 802.11i a malicious NAS can modify the beacon to make the client believe it is connected to a network different from the on the client is actually connected to.

In addition, a malicious NAS can force an authentication server into authorizing access by sending an incorrect Called-Station-ID that belongs to an authorized NAS in the network. This could cause the authentication server to believe it had granted access to a different network or even provider than the one the client got access to.

Both attacks can be prevented by implementing channel bindings, because the server can compare the information that was sent to the client, with information it received from the authenticator during the AAA communication as well as the information stored in the policy database.

Authors' Addresses

T. Charles Clancy
Laboratory for Telecommunications Sciences
US Department of Defense
College Park, MD
USA

Email: clancy@ltsnet.net

Katrin Hoeper
National Institute of Standards and Technology
100 Bureau Drive, mail stop 8930
Gaithersburg, MD 20878
USA

Email: khoeper@nist.gov

Full Copyright Statement

Intellectual Property