Network Working Group   Bernard Aboba
INTERNET-DRAFT   Tim Moore
Category: Informational   Microsoft
<draft-aboba-802-context-00.txt>
11 July 2001

A Model for Context Transfer in IEEE 802

Abstract

IEEE 802.1X [13] enables authenticated access to IEEE 802 media, including Ethernet, Token Ring, and 802.11 wireless LANs. Although Authentication, Authorization and Accounting (AAA) support is optional within IEEE 802.1X, it is expected that many IEEE 802.1X Authenticators will function as AAA clients. Behavior of IEEE 802.1X Authenticators acting as RADIUS clients is described in [24].

The IEEE 802 Inter-Access Point Protocol (IAPP), under development within the IEEE 802.11 TgF working group, supports the transfer of context between access points implementing IEEE 802 technology. Rather than attempting to define both the context transfer protocol and the information elements in a single specification, the IAPP protocol provides a framework for specification and allocation of information elements. The separation of mechanism and data has enabled work to

proceed on parallel tracks, with protocol definition occurring separately from the definition of the information elements.

This document describes how IAPP can be used to support transfer of authentication, authorization and accounting (AAA) context between devices supporting IEEE 802.1X network port authentication [13]. It also defines a framework for allocation of the required information elements within IAPP. This specification is currently being developed within the IEEE 802.11 TgF working group and is being presented to the IETF for informational purposes.

1. Introduction

IEEE 802.1X [13] enables authenticated access to IEEE 802 media, including Ethernet, Token Ring, and 802.11 wireless LANs. Although Authentication, Authorization and Accounting (AAA) support is optional within IEEE 802.1X, it is expected that many IEEE 802.1X Authenticators will function as AAA clients. Behavior of IEEE 802.1X Authenticators acting as RADIUS clients is described in [24].

The IEEE 802 Inter-Access Point Protocol (IAPP), under development within the IEEE 802.11 TgF working group, supports the transfer of context between access points implementing IEEE 802 technology. This document describes how IAPP can be used to support transfer of authentication, authorization and accounting (AAA) context between devices supporting IEEE 802.1X network port authentication [13].

In terms of organization, this document first develops a general model for AAA context transfer. Central to the model is the notion of a "correct" context transfer -- a transfer resulting in the same context on the new access point as would have resulted had a AAA conversation been completed.

The circumstances in which "correct" context transfer can be achieved are analyzed -- demonstrating that this can only be achieved in a limited set of circumstances. As a result, it is suggested that context transfer protocols restrict the domain of applicability to scenarios involving a high degree of homogeneity.

For example, layer 2 context transfer solutions are most likely to be successful transferring context within media families, such as IEEE 802. While the IAPP protocol is expected to be used primarily for transfer of context between IEEE 802.11 access points, it is also possible for it to be used to transfer context between access points supporting other IEEE **802 media, such as IEEE 802.15 or 802.16. Where context transfer between** dissimilar media is required, then higher layer homogeneity is needed. This can be achieved, for example, by restricting applicability to access points supporting Mobile IP.

1.1. Terminology

This document uses the following terms:

Authenticator
An Authenticator is an entity that require authentication from the Supplicant. The Authenticator may be connected to the Supplicant at the other end of a point-to-point LAN segment or 802.11 wireless link.

Authentication Server
An Authentication Server is an entity that provides an Authentication Service to an Authenticator. This service verifies from the credentials provided by the Supplicant, the claim of identity made by the Supplicant.

Port Access Entity (PAE)
The protocol entity associated with a physical or virtual (802.11) Port. A given PAE may support the protocol functionality associated with the Authenticator, Supplicant or both.

Supplicant
A Supplicant is an entity that is being authenticated by an Authenticator. The Supplicant may be connected to the Authenticator at one end of a point-to-point LAN segment or 802.11 wireless link.

1.2. Requirements language

In this document, the key words "MAY", "MUST, "MUST NOT", "optional", "recommended", "SHOULD", and "SHOULD NOT", are to be interpreted as described in [3].

2. Context transfer model

In attempting to transfer context between devices, the first task is to understand how "context" is defined, and what the goal of the context transfer is. For the purpose of this document "context" will refer to the set of state defining the service to be provided to the user.

To date, a number of protocols have been proposed for defining and managing services provided on a per-user basis. RADIUS, defined in [4]-[6], is a first-generation protocol for Authentication, Authorization and Accounting (AAA). Diameter is a next generation AAA protocol currently under development. COPS is a protocol used to manage the establishment of Quality of Service (QoS) state.

In each of these protocols, exchanges are used to establish, and possibly to remove, state from devices. In thinking about transfer of context initially established through such protocols, we would like to propose the "Equivalency Principle":

> For context established via protocol exchanges, transfer of context to a new device can be accomplished by transferring the protocol exchanges that created the context on the original device, and processing them on the new device. For such a context transfer to be successful, the the state created on the new device by processing such an exchange MUST be equivalent to the state that would have been created by having the new device engage in a fresh protocol conversation.

For the equivalency principle to be satisifed, it is necessary for the new device to be able to process the protocol exchanges from the old device, and for those exchanges to result in the same state on the new device. This requires that the protocol messages completely describe the context to be created on the device, and that the effect of processing these messages not depend on state that exists uniquely on the old device, but may not exist on the new device.

For example, a protocol message that describes the state to be attained in terms of deltas from a previous state would not be suitable for use in context transfer, since the effect of the protocol message would differ depending on the previous device state. Similarly, if a protocol message were conditionally executed based on dynamic data, such as the number of users on the device, then the message might have a different effect when processed on the new device than its effect on the old device.

To a large extent, AAA protocols meet the criteria, since the desired device state is completely described by the authorizations. Conditional execution, if it occurs, is relatively rare and usually confined to the AAA server.

The set of messages that establish service context differ, depending on the AAA protocol that is being considered. Within RADIUS [4]-[6], service context is only established via an Access-Accept. Access-Reject messages do not establish context since their purpose is to deny access. Similarly, Access-Challenge messages do not establish context since they represent an intermediate stage within the authentication conversation. Since only one RADIUS message (Access-Accept) establishes service context, to re-establish context on a new device, to first order it is only necessary to transfer Access-Accept messages to the new device, and process them as if they were sent by the RADIUS server.

Note that since only one RADIUS message type can establish context, the message type need not be included explicitly, since it is implicit. As a result, devices supporting transfer of RADIUS context need only transfer AVPs, not the entire RADIUS message.

2.1. "Correct" context transfer

Given this model for context establishment, it is worthwhile to examine when the transfer of context between devices produces a "correct" result.

One way to define correctness in a context transfer is that the transfer establishes on the new device the same context as would have been created had the new device completed a AAA conversation with the authentication server. Ideally, a context transfer should only succeed if it is "correct" in this way. If a context transfer were to establish "incorrect" state, then it would be preferred for such a transfer to fail.

Not all AAA and access device configurations are capable of meeting this definition of "correctness". Implicit within our context transfer model is trust between devices engaging in a context transfer. Since the new device will act on the context transfer as though it had been given the service instructions by a trusted AAA server, it is necessary for the new device to trust the old device.

In transfer of context across administrative domains, such a level of trust may not be possible or appropriate. Therefore it is possible for context transfer to fail even in situations where the devices are homogeneous, due to lack of trust between administrative domains.

If the deployment is heterogeneous, then it may also be difficult to meet this definition of correctness. In these situations, AAA servers often perform conditional evaluation, in which the authorizations returned in an Access-Accept message are contingent on characteristics of the AAA client or the user. For example, in a heterogeneous deployment, the AAA server might return different authorizations depending on the type of device making the request, in order to make sure that the requested service is consistent with device capabilities.

If differences between the new and old device would cause the AAA server to send a different set of messages to the new device than were sent to the old device, then a context transfer between the devices cannot be carried out correctly.

For example, if some access points within a deployment support dynamic VLANs while others do not, then attributes present in the Access-Request (such as the NAS-IP-Address, NAS-Identifier, Vendor-Identifier, etc.)

could be examined to determine when VLAN attributes will be returned, as described in [24].

In practice, this limits the situations in which context transfer can be expected to be successful. Where the deployed devices implement the same set of services, it may be possible to transfer context successfullly. However, where the supported services differ between devices, or where some devices require vendor specific attributes, the context transfer may not succeed. For example, RFC 2865, section 1.1 states:

> "A NAS that does not implement a given service MUST NOT implement the RADIUS attributes for that service. For example, a NAS that is unable to offer ARAP service MUST NOT implement the RADIUS attributes for ARAP. A NAS MUST treat a RADIUS access-accept authorizing an unavailable service as an access-reject instead."

Thus, if a device is to process a context transfer in the same way that it would handle a protocol exchange with a RADIUS server, then if the new device is provided with context for an unavailable service, this MUST cause the context transfer to fail.

Such a failure is a "correct" result within our definition. Presumably a correctly configured AAA server would not request that a device carry out a service that it does not implement. This implies that if the new device were to complete a AAA conversation that it would be likely to receive different service instructions than those present in the context transfer. In such a case, failure of the context transfer is the desired result. This will cause the new device to go back to the AAA server in order to receive the appropriate service definition.

Thus in practice, context transfer is most likely to be successful within a homogeneous device deployment within a single administrative domain. For example, where all the devices support IEEE 802.1X, success is possible, as long as the same set of security services are supported. For example, it would not be advisable to attempt to transfer context between an 802.11 access point implementing WEP to an 802.15 access point without security support. The correct result of such a transfer would be a failure, since if the transfer were blindly carried out, then the user would find themselves moved from a secure to an insecure channel. Thus the definition of an "unsupported service" MUST be encompass requests for unavailable security services.

In general, context transfers between media with different service models should not be expected to be successful. For example, attempts to transfer context between cellular devices and 802.11 access points cannot be "correct" within this model, since the cellular devices do not implement the same set of services as 802.11. Therefore, the correct behavior would be for such context transfers to fail, and for the 802.11

AP to pick up the correct service definition by going back to the AAA server. Thus while transfers between dissimilar technologies will require service interruption, subsequent context transfers between IEEE 802 devices are likely to have a higher probability of success.

2.2. Context handling

AAA is not mandatory to implement for IEEE 802.1X Authenticators. The IEEE 802.1X specification provides guidelines for usage of RADIUS [13], a revised version of which can be found in [24]. However, support for other protocols is feasible. Since a IEEE 802.1X Authenticator may support zero or more AAA protocols and implementation of AAA is non-mandatory, an IEEE 802.1X Authenticator cannot be assumed to implement any particular AAA protocol.

Therefore it is important to define a context transfer mechanism that is protocol agnostic. If two devices share support for a given AAA protocol, then the context transfer mechanism should enable the devices to interoperate. One way to accomplish this is to enable the context transfer mechanism to support multiple AAA protocols within the same message. This allows a device that speaks multiple protocols to interoperate with a device that only supports one of them.

Through use of Information Elements, it is possible to support transfer of context for multiple AAA protocols within the same message. It is proposed than a unique Information Element be allocated to each protocol, and that sub-elements be defined within those Information Elements, if required. Assigning only one Information Element per protocol ensures against exhaustion of the IAPP element space, since the number of AAA attributes may be substantial, so that assignment of Information Elements to individual attributes is to be avoided.

The packaging of AAA messages within a single Information Element also enables compatibility with the definition of correctness described earlier. Within IAPP, a device that receives Information Elements that it does not support will ignore those elements, and process those that it does support.

However, as described earlier, our model of context transfer requires that if a device supports a AAA protocol, that transferred AAA messages MUST be processed according to the rules of the protocol. For RADIUS, this implies that the context transfer MUST fail if unavailable services are requested. As a result, individual RADIUS attributes MUST NOT be encoded as Information Elements within IAPP. Rather, they are encoded as sub-elements. This enables the correct processing to occur. While a device may ignore an entire Information Element, once the Information Element is recognized it must be processed in its entirety. Thus, sub-elements are processed via different rules than Information Elements,

and the distinction is critical to the correct operation of IAPP.

Among other things, this approach enables the context transfer operation to be independent of the supported AAA protocol. For example, a device supporting both Diameter and RADIUS could include Information Elements for both protocols. This would enable transfer of context to a new device supporting either protocol.

2.3. Information Element format

Within IAPP, Information Elements have the following structure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Element Identifier        |            Length           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Information...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Element Identifier

   The Element Identifier field is two octets. It identifies the
   enclosed Information Element.

Length

   The Length field is two octets. It encodes the length of the
   Information Element, including the Element Identifier, Length and
   Information fields.

Information

   The Information field is variable length. It encodes the Information
   Element.

AAA sub-elements are encoded within the Information field as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Organization Unique Identifier          |     Type      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Data...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Organization Unique Identifier (OUI)

The OUI is a three octet field, encoding the Organization Unique Identifier. An OUI of zero is used for standardized sub-elements. Non-zero OUIs can be used to support vendor-specific attributes.

Type

The type field is one octet, and represents the AAA protocol type. To date only RADIUS is assigned a Type field (TBD).

Data

The Data field is of variable length, and contains the information to be transferred. For RADIUS this consists of AVPs.

2.4. Usage guidelines for the RADIUS sub-element

As noted earlier, since RADIUS context is established solely by Access-Accept messages, to first order the RADIUS AVPs that may be included within the RADIUS sub-element are those that are allowable for inclusion within an Access-Accept. The two exceptions are accounting attributes: Acct-Authentic and Acct-Multi-SessionId. The attributes allowable for use with transfers of IEEE 802.1X context are described in Appendix A.

Acct-Authentic provides information on the authentication technique that was utilized on the old access point. Acceptable values are RADIUS, Local and Remote. Typically, it does not make sense to transfer context of sessions established by local authentication, so that the new device will wish to understand the authentication status prior to making a decision on accepting the context transfer.

Acct-Multi-SessionId enables linkage of accounting records from related sessions. As described in [24], it is possible to maintain the same Acct-Multi-SessionId as a user moves between devices. To enable this, it is necessary to transfer the Acct-Multi-SessionId between devices.

3. Open issues

There are open issues relating to transfer of the Message-Authenticator and EAP-Message attributes. Assuming that the IAPP protocol provides support for confidentiality, then transfer of an additional integrity check (Message-Authenticator) is not strictly necessary. However, in order to provide strict conformance to the equivalency principle, it may be desirable to provide this attribute as well, to enable the RADIUS client processing logic to be envoked without modification.

Similarly, since the IEEE 802.1X backend state machine is driven purely by the authentication outcome, not by the contents of the EAP-Message attribute, transferring this attribute is not strictly necessary.

4. Security considerations

4.1. Trust issues

Implicit within our context transfer model is trust between devices engaging in a context transfer. Since the new device will act on the context transfer as though it had been given the service instructions by a trusted AAA server, it is necessary for the new device to trust the old device.

In transfer of context across administrative domains, such a level of trust may not be possible or appropriate. Therefore it is possible for context transfer to fail even in situations where the devices are homogeneous, due to lack of trust between administrative domains.

Another implication of the "equivalency principle" is that the context transfer protocol SHOULD provide the same level of security as the AAA protocol whose context is being transferred. For example, where the AAA protocol is using IPSEC to provide confidentiality, it does not make sense for the context transfer protocol to use shared secret-based hiding.

4.2. Confidentiality

AAA protocol messages may include attributes whose contents are confidential. This includes user passwords, encryption keys, or tunnel passwords. In order to transfer these attributes securely, it is necessary to ensure confidentiality. Within our context transfer model, attributes are processed as though they came from the AAA server. As a result, existing AAA security mechanisms are used in order to ensure confidentiality.

This can be accomplished in two ways. As described in [4], RADIUS attributes can be encrypted using the shared secret shared by the new device and the AAA server. Alternatively, if IPSEC is supported, ESP with a non-null transform can be used to provide confidentiality, as described in [23]. In this case, if a shared secret does not exist, then a null shared secret is assumed.

5. References

[1] Blunk, L., Vollbrecht, J., "PPP Extensible Authentication Protocol (EAP)", RFC 2284, March 1998.

[2] Rivest, R., Dusse, S., "The MD5 Message-Digest Algorithm", RFC 1321, April 1992.

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

[4] Rigney, C., Rubens, A., Simpson, W., Willens, S., "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.

[5] Rigney, C., "RADIUS Accounting", RFC 2866, June 2000.

[6] Rigney, C., Willats, W., Calhoun, P., "RADIUS Extensions", RFC 2869, June 2000.

[7] IEEE Standards for Local and Metropolitan Area Networks: Overview and Architecture, ANSI/IEEE Std 802, 1990.

[8] ISO/IEC 10038 Information technology - Telecommunications and information exchange between systems - Local area networks - Media Access Control (MAC) Bridges, (also ANSI/IEEE Std 802.1D- 1993), 1993.

[9] ISO/IEC Final CD 15802-3 Information technology - Tele-communications and information exchange between systems - Local and metropolitan area networks - Common specifications - Part 3:Media Access Control (MAC) bridges, (current draft available as IEEE P802.1D/D15).

[10] IEEE Standards for Local and Metropolitan Area Networks: Draft Standard for Virtual Bridged Local Area Networks, P802.1Q/D8, January 1998.

[11] ISO/IEC 8802-3 Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Common specifications - Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications, (also ANSI/IEEE Std 802.3- 1996), 1996.

[12] IEEE Standards for Local and Metropolitan Area Networks: Demand Priority Access Method, Physical Layer and Repeater Specification For 100 Mb/s Operation, IEEE Std 802.12-1995.

[13] IEEE Standards for Local and Metropolitan Area Networks: Port based Network Access Control, IEEE Draft 802.1X/D11, March 2001.

[14] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

[15] Yergeau, F., "UTF-8, a transformation format of Unicode and ISO 10646", RFC 2044, October 1996.

[16] Aboba, B., Beadles, M., "The Network Access Identifier", RFC 2486, January 1999.

[17] Alvestrand, H. and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

[18] Dobbertin, H., "The Status of MD5 After a Recent Attack." CryptoBytes Vol.2 No.2, Summer 1996.

[19] Atkinson, R., "Security Architecture for the Internet Protocol", RFC 1825, August 1995.

[20] Zorn, G., Leifer, D., Rubens, A., Shriver, J., Holdrege, M., Goyret, I., "RADIUS Attributes for Tunnel Protocol Support", RFC 2868, June 2000.

[21] Zorn, G., Mitton, D., Aboba, B., "RADIUS Accounting Modifications for Tunnel Protocol Support", RFC 2867, June 2000.

[22] Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std. 802.11-1997, 1997.

[23] Aboba, B., Zorn, G., Mitton, D.,"RADIUS and IPv6", Internet draft (work in progress), draft-aboba-radius-ipv6-10.txt, June 2001.

[24] Congdon, P., Et al. "IEEE 802.1X Usage Guidelines", Internet draft (work in progress), draft-congdon-radius-8021x-15.txt, July 2001.

6. IANA Considerations

This specification does not create any RADIUS attributes nor any new number spaces for IANA administration.

Appendix A - Table of Attributes

The following table provides a guide to which attributes are sent and received as part of IEEE 802.1X authentication, and which attributes are considered part of the "context" to be transferred during roaming. L3 denotes attributes that will be understood only by switches or access points implementing Layer 3 capabilities.

802.1X	Context	#	Attribute
X	X	1	User-Name [4]
		2	User-Password [4]
		3	CHAP-Password [4]
X		4	NAS-IP-Address [4]
X		5	NAS-Port [4]
X	X	6	Service-Type [4]
		7	Framed-Protocol [4]
		8	Framed-IP-Address [4]
		9	Framed-IP-Netmask [4]
L3	X	10	Framed-Routing [4]
X	X	11	Filter-Id [4]
X	X	12	Framed-MTU [4]
		13	Framed-Compression [4]
		14	Login-IP-Host [4]
		15	Login-Service [4]
		16	Login-TCP-Port [4]
X	X	18	Reply-Message [4]
		19	Callback-Number [4]
		20	Callback-Id [4]
L3	X	22	Framed-Route [4]
L3	X	23	Framed-IPX-Network [4]
X	X	24	State [4]
X	X	25	Class [4]
X	X	26	Vendor-Specific [4]
X	X	27	Session-Timeout [4]
X	X	28	Idle-Timeout [4]
X	X	29	Termination-Action [4]
X		30	Called-Station-Id [4]
X		31	Calling-Station-Id [4]
X		32	NAS-Identifier [4]
X		33	Proxy-State [4]
		34	Login-LAT-Service [4]
		35	Login-LAT-Node [4]
		36	Login-LAT-Group [4]

802.1X # Attribute

802.1X	#	Attribute
L3	X	37 Framed-AppleTalk-Link [4]
L3	X	38 Framed-AppleTalk-Network [4]
L3	X	39 Framed-AppleTalk-Zone [4]
X		40 Acct-Status-Type [5]
X		41 Acct-Delay-Time [5]
X		42 Acct-Input-Octets [5]
X		43 Acct-Output-Octets [5]
X		44 Acct-Session-Id [5]
X	X	45 Acct-Authentic [5]
X		46 Acct-Session-Time [5]
X		47 Acct-Input-Packets [5]
X		48 Acct-Output-Packets [5]
X		49 Acct-Terminate-Cause [5]
X	X	50 Acct-Multi-Session-Id [5]
		51 Acct-Link-Count [5]
X		52 Acct-Input-Gigawords [6]
X		53 Acct-Output-Gigawords [6]
X		55 Event-Timestamp [6]
		60 CHAP-Challenge [4]
X	X	61 NAS-Port-Type [4]
		62 Port-Limit [4]
		63 Login-LAT-Port [4]
X	X	64 Tunnel-Type [20]
X	X	65 Tunnel-Medium-Type [20]
L3	X	66 Tunnel-Client-Endpoint [20]
L3	X	67 Tunnel-Server-Endpoint [20]
L3	X	68 Acct-Tunnel-Connection [21]
L3	X	69 Tunnel-Password [20]
		70 ARAP-Password [6]
		71 ARAP-Features [6]
		72 ARAP-Zone-Access [6]
		73 ARAP-Security [6]
		74 ARAP-Security-Data [6]
		75 Password-Retry [6]
		76 Prompt [6]
X		77 Connect-Info [6]
X		78 Configuration-Token [6]
X		79 EAP-Message [6]
X		80 Message-Authenticator [6]
X	X	81 Tunnel-Private-Group-ID [20]
L3	X	82 Tunnel-Assignment-ID [20]
X	X	83 Tunnel-Preference [20]
		84 ARAP-Challenge-Response [6]
802.1X	#	Attribute

802.1X	Context	#	Attribute
X		85	Acct-Interim-Interval [6]
X		86	Acct-Tunnel-Packets-Lost [21]
X		87	NAS-Port-Id [6]
		88	Framed-Pool [6]
L3	X	90	Tunnel-Client-Auth-ID [20]
L3	X	91	Tunnel-Server-Auth-ID [20]
X		TBD	NAS-IPv6-Address [23]
		TBD	Framed-Interface-Id [23]
L3	X	TBD	Framed-IPv6-Prefix [23]
		TBD	Login-IPv6-Host [23]
L3	X	TBD	Framed-IPv6-Route [23]
L3	X	TBD	Framed-IPv6-Pool [23]
802.1X	Context	#	Attribute

Key
===

802.1X = Allowed for use with IEEE 802.1X
Context = Transferred during roaming if available
L3 = implemented only on switches/access points with Layer 3 capabilities

Acknowledgments

The authors would like to acknowledge Bob O'Hara of Informed Technology and Dave Bagby of 3Com for contributions to this document.

Authors' Addresses

Bernard Aboba
Tim Moore
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

EMail: {bernarda, timmoore}@microsoft.com
Phone: +1 425 882 8080
Fax: +1 425 936 7329

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any

effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

Full Copyright Statement

Expiration Date

This memo is filed as <draft-aboba-802-context-00.txt>, and expires January 16, 2002.