Using IEEE 802.1x to Enhance Network Security

WHITE PAPER
802.1X

Table of Contents

Introduction 2
Terms and Technology....2
Understanding 802.1x 3
Introduction....3
802.1x Authentication Process....3
Before Authentication....3
Authentication Process 4
Foundry's Implementation of 802.1x....5
Foundry's 802.1x Implementation and Advantages....6
sFlow Integration....6
802.1x Port Control 6
Re-authenticate....7
Quiet Period....7
EAP-Request/Identity Retransmission Interval....7
Multi-host 7
Single-host....8
RFC Compliance....8
Foundry with 802.1x 8
802.1x integration with IronView Network Manager (INM) 9
Overview....9
How it Works 9
How to Enable 802.1x with sFlow 9
Deployment Guidelines 9
Unsupported Clients....9
Latency Considerations 9
802.1x Considerations with Voice over IP (VoIP) 10
Summary....11

WHITE PAPER
802.1X

INTRODUCTION

The purpose of this document is to introduce IEEE 802.1x standard for enhancing network security, outline the implementation of 802.1x within Foundry's range of products and offer deployment guidelines. The concept of IEEE 802.1x is to provide a standardized security authentication method for IEEE 802 based network technologies, including Local Area Networks (LANs) and Wireless LANs (WLANs).

Whilst technologies such as MAC filtering and Access Control Lists (ACLs) are used to enhance overall network security, the IEEE 802.1x specification provides another level of overall network protection:

- *MAC filtering and ACLs* assume that the administrator has an understanding of what devices and traffic that should be allowed within the network. While this can be achieved in limited scope, it is often too difficult to deploy on a large-scale infrastructure. Most often, ACLs are used in core / data center applications, and MAC filtering is deployed in potentially high-risk network edge connections. This unfortunately does not provide the comprehensive protection many network administrators are seeking.

- *IEEE 802.1x* is a new technology that provides almost unlimited scalability with minimal administration overhead. By authenticating user access at the network edge, network administrators can be assured that no unauthorized access will take place, and all of the user authentication can take place on a centralized authentication server.

TERMS AND TECHNOLOGY

The following are a list of terms and technologies used within 802.1x:

Supplicant (Client) – is the network access device requesting LAN services. Clients that support 802.1x include:

Vendor	Client	Operating System	Foundry Tested
Microsoft	Native to OS	Windows XP	Yes
Funk Software	Odyssey	Windows XP	No

Authenticator – is the network access point that has 802.1x authentication enabled. This includes LAN switch ports and Wireless Access Points (WAP).

Authentication Server – is the server that performs the authentication, allowing or denying access to the network based on username / password. The 802.1x standard specifies that Remote Authentication Dial-In User Service (RADIUS) is the required Authentication Server that supports the following RFC's:

- RFC 2284 PPP Extensible Authentication Protocol (EAP)
- RFC 2865 Remote Authentication Dial In User Service (RADIUS)
- RFC 2869 RADIUS Extensions

EAP – is the protocol that is used between the client and the authenticator. The 802.1x standard specifies encapsulation methods for transmitting EAP messages so they can be carried over different media typed. These include, but are not limited to:

- EAP Over LAN (EAPOL)
- EAP Over Wireless (EAPOW)

Port Access Entry (PAE) – is the 802.1x "logical" component of the client and authenticator that exchange EAP messages.

UNDERSTANDING 802.1X

Introduction

The IEEE 802.1x is a standardized method for securing network access from the network devices. Traditionally, network security has predominantly been the domain of network Servers and clients, based on login authentication to specific resources. If a network user wanted access to network server resources (file and print), then a login challenge needed to be successfully completed. In the case of an all Microsoft domain based network, then the login request is presented to the user when they started up their PC. The PC login username / password would be authenticated against a domain controller, and if successful then the user would be granted access to file & print services specified by the network administrator. This kind of client / server authentication is often used for other network services including email, intranet and other specialized applications

Although client / server authentication is a proven method for securing network resources, it does not provide a total "network" security mechanism that will deter unauthorized access. To counter this, many network equipment vendors have implemented other network-based administrative solutions, including VLANs, Access Control Lists (ACLs) and Media Access Control (MAC) locks. All of which are an effective mechanism for securing network access, but can be administrator intensive depending on what the security criteria is. Each control provides a unique advantage, but are often unique to each vendor. By complimenting the existing network security methods with 802.1x, administrators can be confident that their network perimeter (edge access) is completely secure, as well as having the confidence with interoperability amongst multiple vendors by deploying an IEEE based standard for security.

Since 802.1x is only a perimeter security technology, network administrators should continue to deploy existing security policies to control network traffic:

- 802.1x will deny unauthorized network access, but it will not control network traffic from authorized users. This may be a concern for network administrators that want to secure specific network areas with the use of existing methods including VLANs, ACL's or MAC filtering where it is required.

802.1x Authentication Process

Once 802.1x authentication is enabled (both in the client and authenticator), a successful authentication must be completed before ANY traffic is allowed to transit the network from the client, including critical traffic like DHCP requests regardless of whether link is established between the client and authenticator (switch port).

Before Authentication

To ensure that no unauthorized traffic is transmitted, before successful authentication, the authenticator's PAE is set to uncontrolled. That means that the only messages that will be accepted from the client is EAP requests which will be forwarded to the Authentication server – see figure 1

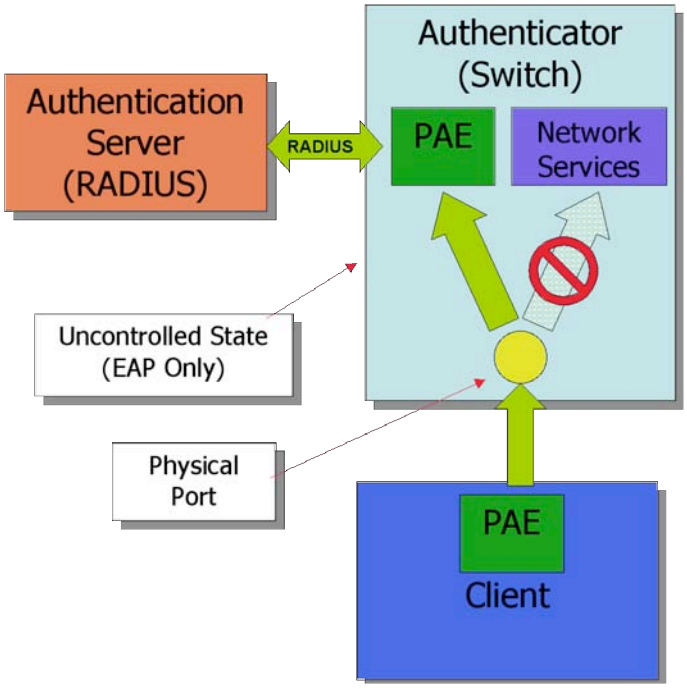

Figure 1 - Before Authentication

Authentication Process

Once activated (powered on and connected to the switch), the 802.1x client will transmit the appropriate EAP message to the authenticator (switch port). The switch port with 802.1x authentication enabled is set to an uncontrolled state, accepting only EAP messages (all other traffic will be discarded). Upon receipt of the clients EAP message, the switch will forward the request to the authentication (RADIUS) server without changing its contents. Although the EAP contents are not changed, the encapsulation must be translated from the originating EAP message to a RADIUS request, therefore the only supported RADIUS servers are ones that support EAP (see RFC Compliance).

Upon receipt of the RADIUS message, the authentication server will grant or deny access to the network. An RADIUS response will then be transmitted back to the switch, which will determine whether the port remains in an uncontrolled state (access denied), or changes to a controlled state (access granted).

If the authentication fails, the authenticator (switch port) will remain in an uncontrolled state, and in some cases the port will be disabled (depends on vendor implementation).

Although the client is the device that requires authentication, either the client or switch can initiate the process. This is determined by timers which are set in both the client and the switch.

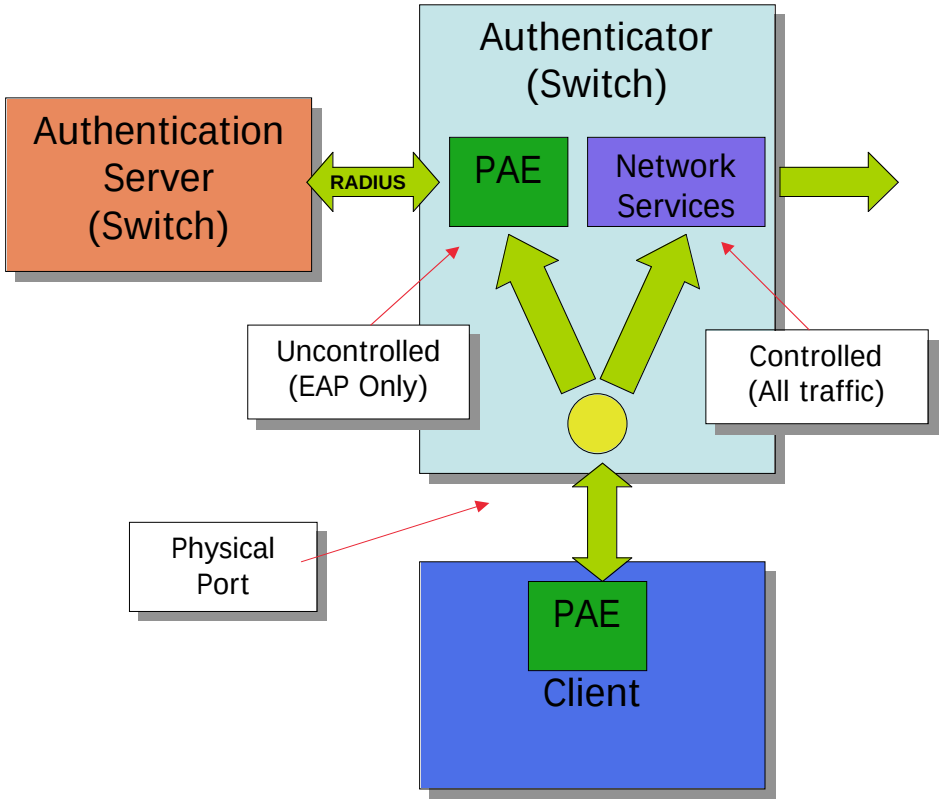

Figure 2 - After successful authentication

FOUNDRY'S IMPLEMENTATION OF 802.1X

By default, all switch ports within Foundry products are placed in the authorized state, allowing all traffic to pass through (traditional method used for all Foundry products). When 802.1x authentication is activated, the interface port is placed initially in the unauthorized (uncontrolled) state, which will only accept 802.1x authentication requests. When successful authentication is complete, the port is then placed in the authorized (controlled) state until the Client logs off. Once the client logs off, then a logoff exchange process is executed, alerting the switch port that is must change to the unauthorized state.

The following diagram shows the message exchange between the client, switch and server, for the authentication process. In this example, we show that the switch initiated the authentication process.

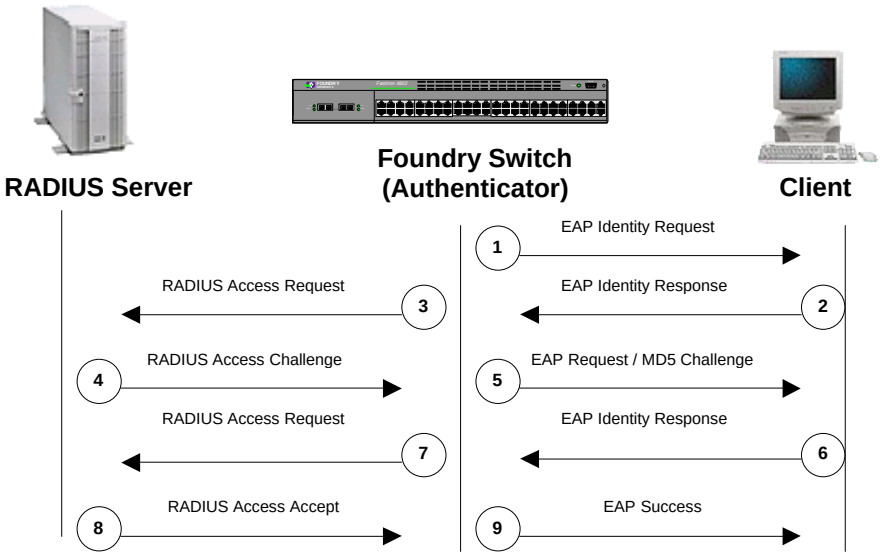

Figure 3 - EAP / RADIUS Message Exchange

In this example, the Authenticator (the Foundry device) initiates communication with an 802.1X-enabled Client. When the Client responds, it is prompted for a username and password. The Authenticator passes this information to the Authentication Server, which determines whether the Client can access services provided by the Authenticator. When the RADIUS server successfully authenticates the Client, the port is placed in the authorized state. When the Client logs off, the port is placed in the unauthorized state again. If the Client does not support 802.1X, authentication cannot take place. The Foundry device sends EAP-Request/Identity frames to the Client, but the Client does not respond to them. When a Client that supports 802.1X attempts to gain access through a non-802.1X-enabled port, it sends an EAP start frame to the Foundry device. When the device does not respond, the Client considers the port to be authorized, and starts sending normal traffic.

Foundry's 802.1x Implementation and Advantages

sFlow Integration

Feature: Incorporates support for the standards-based sFlow network monitoring technology. By taking advantage of username identification available within the 802.1x authentication process, we can deliver username based traffic flow statistics and reports. See product matrix below for supported platforms.

Benefit: This provides administrators with the ability to determine which user(s) may be responsible for network problems, or experiencing network issues without having to manually match username with network or MAC address, significantly reducing troubleshooting efforts and enhancing overall network monitoring.

802.1x Port Control

Feature: Used to enable 802.1x port authentication, requiring the client be 802.1x enabled, and a RADIUS server configured. By default, all ports are set to the authorized state, allowing all traffic to pass through the switch port without any authentication

requirements. When 802.1x is implemented, the port is set to the unauthorized mode (see Figure 1), allowing only EAP traffic to pass through the switch port. After a successful authentication, the switch port is placed in authorized mode allowing all traffic to pass through the port (see Figure 2).

Benefit: Allows network administrators to provide network access security based on industry standard technologies.

Re-authenticate

Feature: Allows network administrators to determine a time period where an 802.1x client will be required to re-authenticate.

Benefit: This ensures that once a port is authorized, it cannot be compromised by a non-802.1x client. This could be possible is the user were to attach a hub to the switch port and authenticate with the 802.1x client. Another computer (non-802.1x) could be connected to the hub and gain access to the network. By default, re-authenticate is not set when 802.1x is enabled, and must be set manually. The default time period for re-authenticate is 3600 seconds (recommended), but can be modified from 1 to 4294967295 seconds.

Quiet Period

Feature: Is specified as the time the authenticator (Foundry switch) will wait, if the client is not successfully authenticated, before allowing the client to try again. The default is set to 60 (recommended) seconds, but can be modified between 0 and 4294967295.

Benefit: This ensures that the client can attempt to re-login if the original login request was incorrect. Rather than locking the port out completely, the user can re-attempt authentication after the pre-defined period, minimizing administrator intervention from failed authentication.

EAP-Request/Identity Retransmission Interval

Feature: When the Foundry device sends a Client an EAP-request/identity frame, it expects to receive an EAP-response/identity frame from the Client. If the Client does not send back an EAPresponse/identity frame, the device waits a specified amount of time and then retransmits the EAP-request/identity frame. The default is set to 30 seconds (recommended), but can be modified between 0 and 4294967295.

Benefit: Allows the to continue with the authentication process if the client was temporarily unable to respond to original request.

Multi-host

Feature: When the port has 802.1x security enabled, it can be configured with multi-host support. This allows more that one client (other than the originally authenticated client) to access the switch port.

Benefits: Allows multiple devices in a shared media (hub-like) infrastructure to access the network

Single-host

Feature: When the port has 802.1x security enabled, the security mode is set to single-host by default. This mode is used to identify the originally authenticated station. Once the 802.1x client is authenticated, the switch port is programmed with the PC's MAC address, and will only allow traffic into the network from that particular MAC address.

Benefits: This feature enforces security for all user access points, overcoming the possibility of a user connecting a shared-media device (like a hub), which would permit traffic from unauthorized devices.

RFC Compliance

Foundry's 802.1x implementation complies with the following RFC's:

- RFC 2284 PPP Extensible Authentication Protocol (EAP)
- RFC 2865 Remote Authentication Dial In User Service (RADIUS)
- RFC 2869 RADIUS Extensions

Foundry's Products With 802.1x Implementation

Foundry's switching products will provide support for 802.1x with the 7.6.01 release of software, including Switch, base Layer 3, and Full Layer 3 code. Below is a product matrix including product name and specific software release required.

Product Name	Software version	sFlow Integration
FastIron Edge 2402	**2.0**	**Yes**
FastIron Edge 4802	**2.0**	**Yes**
FastIron Edge 9604	**2.0**	**Yes**
FastIron II	7.6.01	No
FastIron II Plus	7.6.01	No
FastIron III	7.6.01	No
FastIron 400	**7.6.01**	**Yes**
FastIron 800	**7.6.01**	**Yes**
FastIron 1500	**7.6.01**	**Yes**
BigIron 4000 (IronCore)	7.6.01	No
BigIron 8000 (IronCore)	7.6.01	No
BigIron 15000 (IronCore)	7.6.01	No
BigIron 4000 (JetCore)	**7.6.01**	**Yes**
BigIron 8000 (JetCore)	**7.6.01**	**Yes**
BigIron 15000 (JetCore)	**7.6.01**	**Yes**

802.1x integration with IronView Network Manager (INM)

Overview

Foundry's INM v1.6 integrates 802.1x username identification with the sFlow collection reports. By enabling 802.1x identification, administrators can easily identify users with specific traffic flows, eliminating difficult address (IP or MAC) to user matching.

How it Works

Once the switch port is configured with 802.1x authentication enabled, the switch stores the user identification (userid) from each authentication request. Once the user userid has been stored, sFlow will take a copy of that information, and export it with the usual sFlow update packets for that port.

How to Enable 802.1x with sFlow

Exporting 802.1x information within sFlow is automatic. You simply need to have the switch port configured with 802.1x authentication support, and sFlow needs to be enabled. Once these two features are enabled, and INM 1.6 is used as the collector, you will see an 802.1x entry for each traffic flow.

Deployment Guidelines

Although 802.1x is a network edge authentication technology, some considerations must be taken into account. Once the decision has been made to deploy this feature, administrators must determine which switches, and specific ports will need to be enabled with 802.1x, based on client support.

Unsupported Clients

Some considerations when deciding whether to enable 802.1x on specific switch ports are: what devices do not support 802.1x authentication, or, what devices have their own 802.1x implementation. Below are some examples.

- Devices that do not support 802.1x may include printers, servers, Voice over IP (VoIP) phones and call processing servers, as well as other peripherals. In these examples, 802.1x should not be enabled on the switch port, as the device would not appear on the network since it cannot authenticate.
- Devices that have their own 802.1x implementation include Wireless Access Points (WAPs). In this example, the WAP provides authentication for the wireless clients, and the switch port the WAP is attached to should not have 802.1x enabled, otherwise the WAP would not be able to communicate with the network and all clients would be denied access.

Latency Considerations

Another consideration when implementing 802.1x is latency, particularly with relation to DHCP timeouts. This needs to be tested, and understood with each specific network design. Since the 802.1x authentication process is the first process undertaken by the client, it may effect the DHCP process. For example, if the 802.1x request / response

time exceeded the DHCP timeout, then the client would be authenticated to the network, but may not receive an IP address to participate on the network. Unfortunately there are no specific guides to ensure this will never happen, as it is a function of network latency and the RADIUS server response times during the 802.1x authentication.

802.1x Considerations with Voice over IP (VoIP)

There are some considerations that must be understood when deploying VoIP in conjunction with 802.1x. The most important consideration is to do with the IP phone characteristics, particularly when using "dual-port" IP phones (see Dual-port IP phone connection diagram). The following are two examples of dual-port IP phone behavior when they connect to the network.

Figure 4 - Dual-port IP Phone Connection

Example 1

In this example, the IP phone requires a completed DHCP and IP PBX registration before it will allow any traffic to pass from the PC client to the network. This could cause a concern for users if the IP phone does not support 802.1x. In this case, the IP phone will initially transmit DHCP requests, looking for an IP address (and it will not allow the EAP requests from the client PC to pass into the network switch port). Since the network switch port is only allowing EAP requests into the network, it will not pass the DHCP request from the IP phone, which will cause the IP phone to fail bootup. If this happens, the PC will never get to authenticate with the switch, causing the PC to stay disconnected from the network.

Example 2

In this example, the IP phone will allow any traffic to pass from the PC client to the network regardless of whether the Phone has completed it's bootup or PBX registration. Once the PC client has completed authentication, network traffic is now able to pass through the network. To ensure that the IP phone can complete its bootup and registration, the switch port must be set to multi-host support when enabling 802.1x. Otherwise, the PC will be the only device allowed to use this network port, and all of the IP phone traffic will be denied.

WHITE PAPER
802.1X

Summary

By implementing IEEE 802.1x, administrators can protect the network from unauthorized user access, minimizing potential security breaches including Denial of Service (DoS) attacks within their network infrastructure. Although implementing 802.1x provides enhanced network edge security, it is important for network administrators to plan and deploy this technology based on their requirements, taking into consideration devices like VoIP phones and Wireless Access Points.

It is important to note that 802.1x technology is complimentary with existing security technologies, and is not a replacement. Since 802.1x can be considered to be a perimeter security measure, there will still be the need for network security techniques like VLANs and ACLs.