802.1X
DEPLOYMENT GUIDE

Version 1.0

May 19, 2014

LEGAL NOTES

REGARDING TRADEMARKS

The features described in this guide vary depending on your device model.

VERSION HISTORY

Version	Date	Notes
1.0	5/19/2014	Release official

TABLE OF CONTENTS

DOCUMENT SCOPE

WHAT IS THE PURPOSE OF THIS DOCUMENT?

This document is designed to be a resource for deploying a wired 802.1X environment for TA Triumph-Adler/UTAX MFPs. It will help the reader understand what 802.1x is and how it can be used. It includes some background and general information regarding 802.1X and the supporting components. The focus, however, is on TA Triumph-Adler/UTAX MFPs and their configuration for an 802.1X deployment.

There are many different options available when deploying 802.1x. Most of these will be defined by the existing infrastructure in your organization. For example, the authentication server will dictate which EAP methods are possible to use. This document has included all of the options available based on the features of TA Triumph-Adler/UTAX MFPs. Any methods that are not available to TA Triumph-Adler/UTAX MFPs will not be covered in this document.

Each major section covers one of the three main elements of 802.1x: the **supplicant**, the **authenticator**, and the **authentication** server. To that end, this document is modular in design. It does not need to be read from beginning to end.

The following topics are included:

- **IEEE 802.1x overview and introduction**
  - ✓ This describes what 802.1x is for and how it works.
- **TA Triumph-Adler/UTAX MFP configuration for 802.1x**
  - ✓ This includes MFPs running CCRX or Command Center.
- **Network switch configuration for 802.1x and RADIUS**
  - ✓ This has an example using a Cisco 2960 switch.
- **Authentication server configuration for 802.1x and the supported EAP methods**
  - ✓ This includes Windows 2008 and 2012, FreeRADIUS on Ubuntu 12.04, and Cisco ACS 4.1.

This guide should provide enough information for users new to 802.1x to grasp the fundamentals. It will aide in configuring a basic infrastructure as well. It is encouraged for users attempting to integrate 802.1x into their environments to read any additional materials provided. References for more advanced instruction will be provided when relevant.

WHAT DOES THIS DOCUMENT NOT INCLUDE?

- This document is not a reference for network security design or security best practices.
- It is not a fully inclusive walkthrough for a production-ready 802.1x deployment.
- Wireless 802.1x deployments are not referenced, although most of the topics will apply.
- It does not include all EAP methods, only those supported by TA Triumph-Adler/UTAX MFPs. However, the methods supported are the most popular and widely supported in the industry.

802.1X OVERVIEW

IEEE 802.1x is a standard for port-based network access control. It provides the capability to permit or deny network access based on the identity and credentials of the user or device attempting to connect to the network. This capability prevents unauthorized devices from accessing the network after connecting to a switch port.

An 802.1x deployment is one method of preventing unwanted access to a network. Without 802.1x, any device could connect into any available network port. This could be in an unused cubicle or a vacant conference room. This device could then acquire a network address and start communicating with other devices on the network. It could also silently monitor network traffic to collect information. Either way, it is allowed to access the network with no knowledge of the administrators.

With 802.1x enabled, that same device would have to successfully authenticate before it is granted access to the network. If the unknown device does not provide the correct credentials, the port would remain shutdown. Advanced configurations could shutdown that network port permanently and send alerts to administrators. Alternatively, the port could be set to access a specific VLAN that has limited access for unauthenticated guests. There are many options that 802.1x can provide for network security, management, and accounting.

802.1x is commonly used in wireless networks to prevent unwanted users from connecting to it. While there are some differences between a wireless and wired deployment, the overall concepts are the same. This document will focus only on wired networks.

The complete IEEE 802.1x specification can be downloaded here:
http://standards.ieee.org/getieee802/download/802.1X-2010.pdf

PRIMARY COMPONENTS TO 802.1X

802.1x has its own terminology for some of the required devices. The three primary components are

- **Supplicant**
- **Authenticator**
- **Authentication server**.

These are described below along with a few other items found in a typical 802.1x deployment. Additional terms and concepts related to 802.1x are also included.

SUPPLICANT: A supplicant is any device that is requesting network access when connected to a switch port. Supplicants must have software that runs 802.1x to be able to request access. It is common to refer to the actual software as the supplicant. The supplicant's 802.1x software allows it to communicate, using EAP or EAPOL, to the Authenticator. A device that does not contain the supplicant software is technically not an 802.1x supplicant.

Examples: MFPs, PCs, laptops

AUTHENTICATOR: The authenticator is the device that acts as a translator between the supplicant and authentication server. It takes the EAP frames from the supplicant and encapsulates them in RADIUS packets to forward to the authentication server. Authenticators must support 802.1x and have ports enabled for

802.1x operation (not all devices do). Most business or enterprise class switches will have an authenticator and support for 802.1x.

Example: Switches, wireless APs

AUTHENTICATION SERVER: The authentication server is the device that decides whether the supplicant is granted access. The authenticator forwards all requests to the authentication server and expects a reply. The reply will permit or deny access to the connected supplicant. Optionally, the authentication server could provide additional information, such as amount of time access is granted or VLAN configuration. Additionally, the server could also collect accounting information that can be logged and viewed by administrators.

Examples: Windows Network Policy Server, FreeRADIUS, Cisco ACS

OTHER TERMS AND CONCEPTS

BACKEND USER DATABASE: A central storage of credentials that the authentication server can use to validate connection attempts. This can exist on the authentication server or as an external entity.

Examples: Microsoft Active Directory, SQL, LDAP, Cisco ACS local database

PUBLIC KEY INFRASTRUCTURE (PKI): PKI is a complex system that is used to manage X.509 certificates (also known as SSL certificates) and their related keys. A PKI is used when digital certificates are required to provide authentication for servers, devices, or users. A PKI is meant to provide a level of trust between devices communicating on a network.

A PKI will have one (or more) servers known as CA (certificate authority) servers that can issue certificates. The CA server also has a certificate known as the CA root certificate. This certificate is typically provided to all clients and network devices within an organization. A PKI can be internal to an organization or be created with public CA servers. Public certificates cost money to acquire.

The details of a PKI are beyond the scope of this document. When referenced in this document, a PKI is assumed to be functional and be able to issue the required server and client certificates as needed. This is a requirement for EAP-TLS since server and client certificates are needed.

Note: Some smaller organizations may not have a PKI in place. To use EAP-PEAP, EAP-TTLS, or EAP-FAST, the authentication server can act as a CA or simply use a self-signed certificate. If a self-signed certificate is used, it must be deployed to the MFP and installed as a root certificate.

PERSONAL INFORMATION EXCHANGE (PFX): A .PFX is a file format that contains a variety of objects. It will commonly include a certificate, the private key, and the CA root certificate(s). These are used to import the client and root certificates into the MFPs or authentication servers.

EXTENSIBLE AUTHENTICATION PROTOCOL (EAP)

The **Extensible Authentication Protocol** (**EAP**) is an authentication framework that supports multiple authentication methods. It was designed for use in network access authentication where the IP layer may not be available. On wired implementations, EAP is also referred to as EAP over LAN (**EAPOL**). EAP is used by the supplicant and authenticator to communicate.

Keep in mind the following points concerning EAP:

- EAP is a transport for authentication and not the authentication itself. For example, EAP-TLS uses EAP to carry the information to conduct the TLS negotiation.
- EAP must be supported on the supplicant and authenticator.
- EAP does not require IP to operate, as it is a Layer-2 protocol.
- As a Layer-2 protocol, there are no UDP/TCP ports for EAP.

EAP FORMAT

An EAP frame consists of the following fields. See Figure 1.

- **Code**: identifies the type of EAP packet
  - **1** = Request
  - **2** = Response
  - **3** = Success
  - **4** = Failure
- **Identifier**: used to match responses and requests.
- **Length**: indicates the length of the EAP packet, including header and data.
- **Data**: contains the data of the EAP frame, such as the TLS negotiation.

Figure 1 - EAP Frame Fields

Code (1 byte)	**Identifier** (1 byte)	**Packet Length** (2 bytes)
Data (0+ bytes)		

The EAP Type determines the nature of the EAP frame. There are a variety of EAP Types available. Some example EAP Types are:

- EAP-TLS (**Type 13**)
- EAP-TTLS (**Type 21**)
- EAP-PEAP (**Type 25**)
- EAP-FAST (**Type 43**)

More information on EAP Types and Codes can be found here:
http://www.iana.org/assignments/eap-numbers/eap-numbers.xhtml

EAP RFCS

- RFC 3748
- RFC 5247

- RFC 5269

RADIUS

RADIUS (Remote Authentication Dial-In Service) is an authentication and accounting network protocol. In 802.1x deployments, the RADIUS protocol is used to communicate authentication information between the authenticator and the authentication server.

The RADIUS protocol uses the client-server model. As far as RADIUS is concerned, the client is the 802.1x authenticator and the server runs the RADIUS software. The server software provides the AAA (Authentication, Authorization, and Accounting) architecture needed for 802.1x or other network services. The AAA servers are commonly referred to as RADIUS servers.

Note: Keep in mind that RADIUS is a network protocol as well as a AAA server. Most authentication servers are referred to as RADIUS servers.

More RADIUS information can be found here:
http://networkradius.com/doc/FreeRADIUS%20Technical%20Guide.pdf

RADIUS SYSTEM COMPONENTS

Just as 802.1x has its own terms, so does RADIUS.

- **User / Device** - Requests access to the network. This is the 802.1x supplicant.
- **Network Access Server (NAS)** - Provides access to the network. This is the 802.1x authenticator.
- **Authentication Server** - Authenticates requests from the NAS. This is the 802.1x authentication server.
- **Data Store** - An optional database with user information.

RADIUS FORMAT

RADIUS uses UDP as the transport protocol. The port numbers are as follows:

- **UDP 1812/1813** - these are the officially assigned port numbers (server port/accounting port).
- **UDP 1645/1646** - these were used in early deployments of RADIUS (server port/accounting port).

A RADIUS packet consists of the following fields. See Figure 2.

- **Code**: identifies the type of RADIUS packet. Below are some common code examples.
    - 1 = Access-Request
    - 2 = Access-Accept
    - 3 = Access-Reject
    - 11 = Access-Challenge
- **Identifier**: aids in matching requests and replies.
- **Length**: indicates the length of the packet, including the code, identifier, length, authenticator, and attributes fields.
- **Authenticator**: used to authenticate the reply from the RADIUS server, and is used in the password-hiding algorithm.

- **Attributes**: the Attribute Value Pairs (AVPs) carry the specific authentication, authorization, and information details. Refer to the RFC for details on the various Attributes.

Figure 2 - RADIUS Packet Fields

Code (1 byte)	Identifier (1 byte)	Packet Length (2 bytes)
Authenticator (16 bytes)		
Attributes …		

RADIUS RFCS

- RFC 2865
- RFC 2579

TLS

Transport Layer Security (TLS) authenticates and protects network communication by using certificate-based authentication and encryption. The working details of TLS are beyond the scope of this document. However, it is important to understand how TLS is used by each EAP method.

TLS is used by some EAP methods to create a secure connection to protect the authentication process. This means the server will present its certificate to the client for validation. This requires the client to possess the correct CA issued root certificate to be able to validate the server certificate. Without the trusted root certificate, the TLS connection will fail since the client does cannot verify the server.

Keep this in mind when deploying an 802.1x infrastructure. If an existing PKI is in place, the server and client can use certificates issued from the CA server. The authentication server's certificate will need to be trusted by each client, or MFP, to work.

Note: If invalid certificates are used, TLS will fail and so will the 802.1x attempt.

802.1X BASIC COMMUNICATION FLOW

Figure 3 shows a simple logical diagram of the flow of traffic between the major components.

1. The supplicant communicates to the authenticator using EAP frames.
2. The authenticator places the EAP message into an AVP in a RADIUS packet to send to the authentication server.
3. The authentication server responds with an EAP message inside a RADIUS packet.
4. The authenticator removes the EAP message from the RADIUS packet to send back to the supplicant as an EAP frame.

Note: Keep in mind that the EAP communication is performed without a network layer protocol (such as IP) but RADIUS communication is.

Figure 3 - Basic Communication Flow

A more detailed description of the 802.1x communication flow can be found in the 802.1x Communication Detail section below.

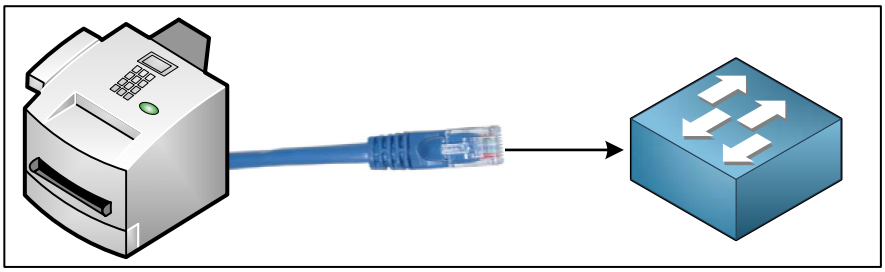
SUMMARY OF 802.1X OPERATIONS

This section provides an overview of the operation of 802.1X. It describes the state of the devices during different stages of the process. It is very high-level that should help understand the 802.1x process and the interaction between the components. For this example, assume that all devices have been properly configured for 802.1x.

CONNECT THE MFP

For this example, the MFP is configured but not connected to the switch. The process is started by connecting the MFP to an 802.1x enabled port on the switch. See Figure 4.

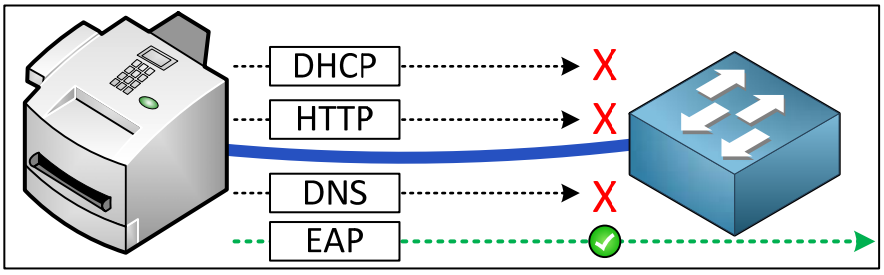
Figure 4 - Connect the MFP



MFP IS NOT AUTHORIZED

Upon connecting, the MFP's network traffic will be dropped by the switch port because it has not yet been authenticated nor authorized for access. As far as the network is concerned, this device is unknown and therefore not allowed to communicate. In this state, the switch will only allow process EAP (or EAPOL) frames. All other traffic is dropped by the switch. For example, DHCP is dropped and the MFP will not be able to configure an IP address until authorized for network access. See Figure 5.

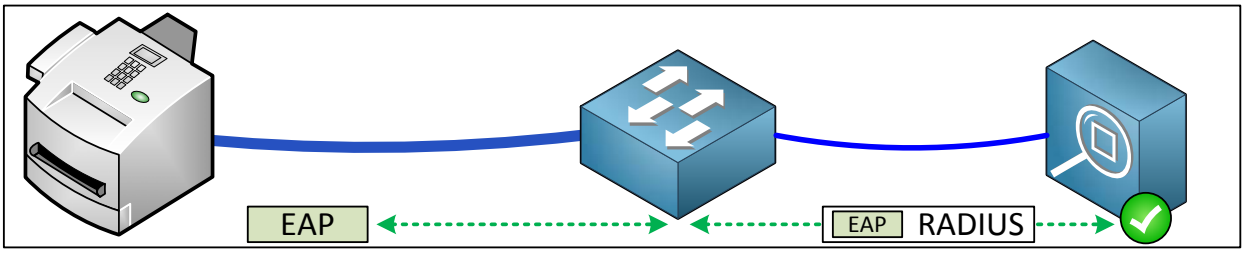
Figure 5 - MFP is not authorized



SUCCESSFUL AUTHENTICATION PROCESS

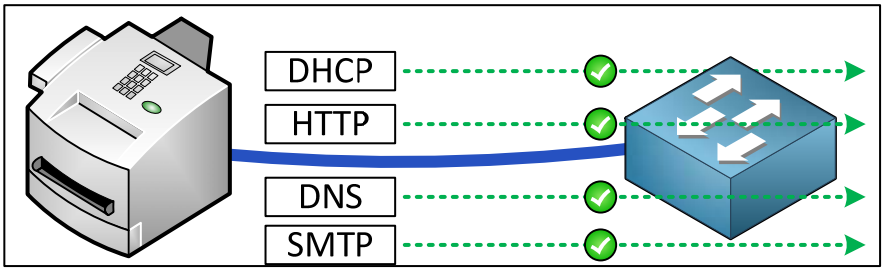
EAP is used for communication between the MFP and the switch. The switch and authentication server communicate using the RADIUS protocol. At this point, EAP frames are still the only traffic allowed by the switch. The switch sends the EAP information to the server using RADIUS packets. See Figure 6.

Figure 6 - Authentication process is successful



The server responds by accepting or rejecting the authentication. If the authentication is successful, the server will tell the switch to allow the device access to the network by unblocking the port. All of the MFP's traffic is now allowed on the network. Additionally, other policies can be implemented such as time limits, VLAN assignment, and access lists. See Figure 7.

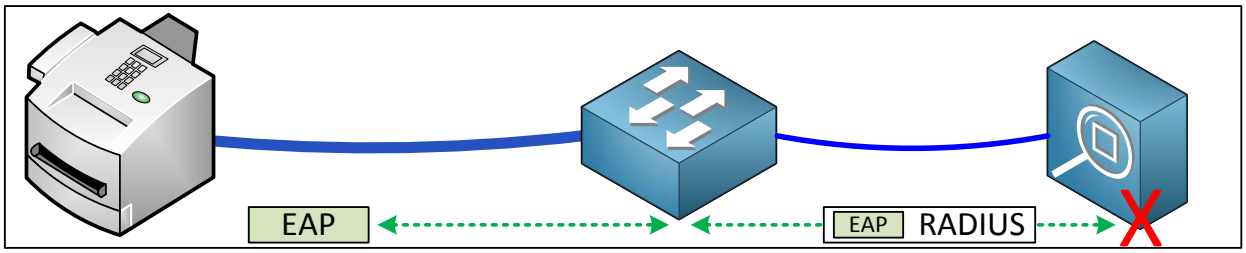
Figure 7 - MFP is authenticated



FAILED AUTHENTICATION PROCESS

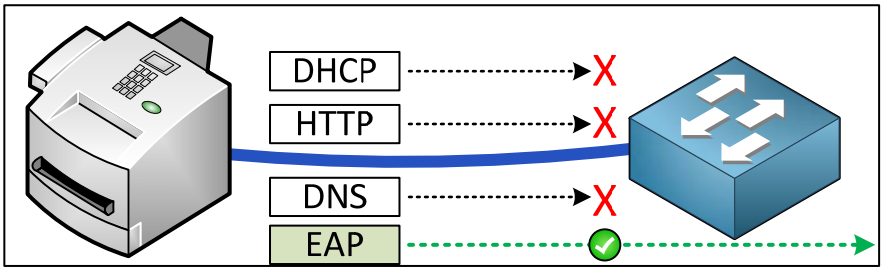
If the authentication fails, perhaps due to bad credentials, the server informs the switch of the failure and the port remains blocked. See Figure 8.

Figure 8 - Authentication process fails



No traffic, except EAP, will be allowed. See Figure 9. Depending on the server and switch settings, the device may try to authenticate again or the port could be shut down and a network administrator could be notified.

Figure 9 - MFP not authenticated



DETAILED 802.1X OPERATIONS

The example in the previous section can be broken down into steps that are more detailed. These steps are explained below.

802.1X HIGH-LEVEL STEPS

STEP 1: INITIATION

Either the MFP or switch can start the 802.1X authentication process.

SWITCH INITIATION:
By default, the switch will periodically send **EAP-Request-Identity** frames out of each 802.1X enabled port. If the MFP has 802.1x enabled, it will respond with its identity and the authentication process can begin. If the MFP does not have 802.1x enabled the port will remain un-authorized. The switch will not accept any traffic other than EAP frames.
Note: A switch may have 802.1x enabled, but not configured on all ports. Confirm that the port to be used has 802.1x enabled.

MFP INITIATION:
The MFP will also transmit **EAPOL-Start** frames when:

1. It is connected to a switch port.
2. The MFPs network service or the device is restarted.

The switch will see the **EAPOL-Start** frame and know an 802.1X device is connected. This will start the 802.1X authentication process without having to wait for the switch to initiate.

Regardless of the device starting the negotiation, the initiation section concludes when the MFP receives and processes an **EAP-Request-Identity** frame from the switch.

STEP 2: AUTHENTICATION AND AUTHORIZATION

After receiving the **EAP-Request-Identity** frame, the MFP will respond with an **EAP-Response** that contains the identity (username). During this stage, the switch relays EAP messages between the MFP and the authentication server. The switch will take the EAP message, place it into an AV-pair within a RADIUS packet, and send it to the server. The server's response will be an EAP message inside a RADIUS packet with the switch as the destination.

The authentication stage can be broken into a two parts:

PART 1: The MFP and the authentication server agree on the EAP method: TLS, PEAP, TTLS, or FAST.

PART 2: Authentication is performed using the selected EAP method. The specifics will vary depending on the method used. See the EAP Methods section for details on each method's authentication process.

The EAP method defines the type of credential to be used to validate the identity of the MFP. It also defines how the credentials are submitted. Depending on the EAP method, the supplicant may submit a password, certificate, token, or other credential. That credential can then be passed inside a TLS-encrypted tunnel, as a hash or in some other protected form.

The server will choose to accept or reject the authentication request. A request may be rejected due to a number of reasons:

- Invalid username/password
- Invalid or expired certificates
- The policy denies user access

If rejected, the server will inform the switch and that port remains in a blocking state. Other tasks may be performed, such as putting the port into an error state, allowing authentication to be retried, or putting the port into a guest VLAN.
Note: Configuring these advanced tasks is outside the scope of this document.

If authentication is successful, the server informs the switch and the port is opened to allow traffic to flow. Additional policies may still be specified such as VLAN assignment or time limits. At this point, protocols such as DHCP will be allowed through the switch and onto the network.

STEP 3: ACCOUNTING

If configured, the switch can send a RADIUS **Accounting-Request** message to the authentication server. This will provide details of the session. This is optional.

STEP 4: SESSION TERMINATION

An 802.1 x session can be terminated in a few ways.

LINK DOWN: 802.1x sessions are immediately terminated when the authenticated device disconnects from the port. When unplugged or powered off, the 802.1x session is immediately cleared to prevent another device from connecting without authorization. When reconnected, the MFP or any other 802.1x device will have to re-authenticate.

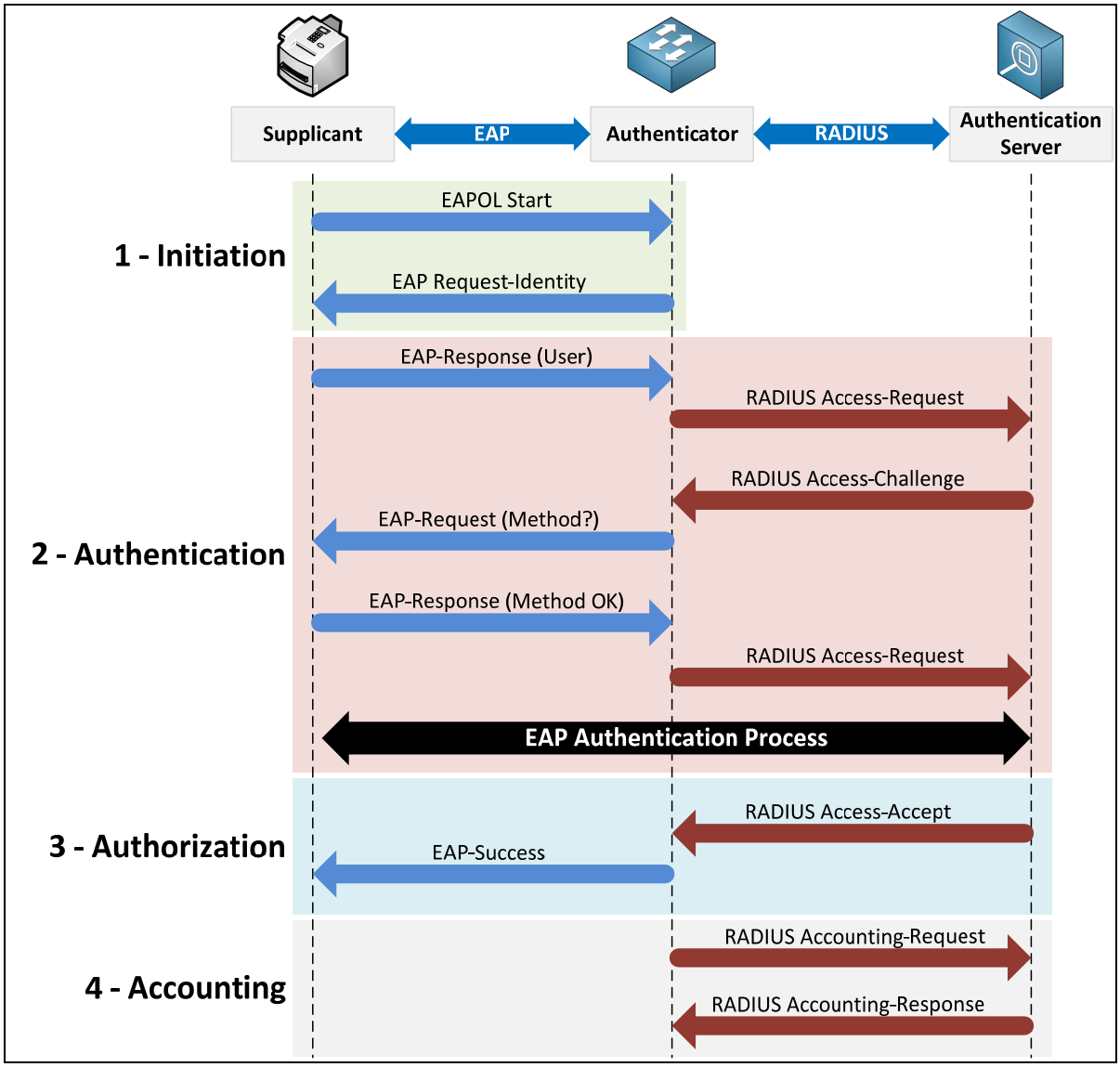
INACTIVITY TIMEOUT: A switch can have an inactivity timer set or receive a timer value from the authentication server. When it expires, the authenticated 802.1x session is removed. Devices that do not transmit data before the inactivity timer expires will have their port disabled since the switch believes there is not device connected. This is useful if the MFP is connected to an intermediate device such as a hub that is not 802.1x capable.

OTHER METHODS: Other methods of session termination include CDP notifications (Cisco products) and EAPOL Logoff messages.

802.1X COMMUNICATION DETAIL

Figure 10 diagrams a more detailed communication flow of an 802.1X authentication session.

Figure 10 - 802.1X communication flow



EAP METHODS

There are four available EAP methods for TA Triumph-Adler/UTAX devices:

- **EAP-TLS**
- **EAP-PEAP**
- **EAP-TTLS**
- **EAP-FAST**

Each method operates differently and requires separate configuration settings on the authentication server and the MFP. The switch configuration is not dependent on the EAP method used. The switch will encapsulate all EAP traffic into RADIUS packets to send to the authentication server. The MFP and the server must be configured to use the same EAP authentication method.

There are pros and cons to each EAP method. When incorporating 802.1x into an existing infrastructure, the decision will likely depend on the existing authentication server. EAP-TLS and EAP-PEAP are the most popular and widely supported methods. Both can be used with Windows and FreeRADIUS servers. EAP-TTLS also has good support but not with Windows server products. EAP-FAST is supported mainly by Cisco products. TA Triumph-Adler/UTAX products support all four methods, depending on the model:

- **CCRX**: All methods are supported.
- **Command Center**: EAP-TLS & EAP-PEAP.

Figure 11 compares the different EAP methods supported by TA Triumph-Adler/UTAX devices. This is not an all-inclusive feature-by-feature comparison between the protocols. It simply highlights their differences when used with TA Triumph-Adler/UTAX devices.

Figure 11 - EAP method comparison

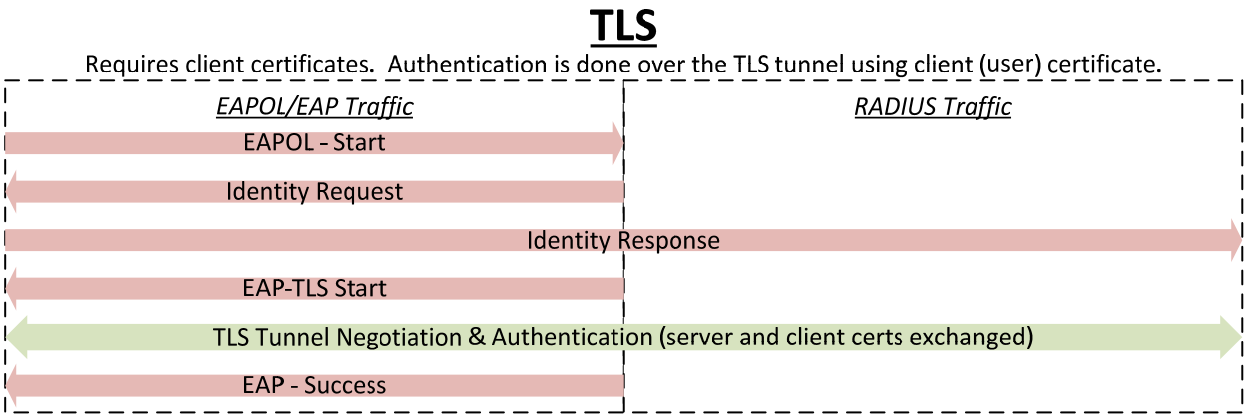
Item	EAP-TLS	EAP-PEAP	EAP-TTLS	EAP-FAST
Client Certificate Required?	Yes	Optional	Optional	Optional
Server Certificate Required?	Yes	Yes	Yes	Yes
Root Certificate Required?	Yes	Yes	Yes	Yes
TLS Tunnel Created	Yes	Yes	Yes	Yes
Authentication Method	Certificate	MSCHAPv2	MSCHAPv2 MSCHAP CHAP PAP	MSCHAPv2
Username/Password	No	Yes	Yes	Yes

The following sections will provide a brief summary of each EAP method. It is recommended to view the referenced RFCs for additional details.

EAP-TLS

EAP-TLS (EAP-Transport Layer Security) is a widely supported, deployed, and secure EAP method. While it offers excellent security, it also requires a more complicated installation due to the requirement for mutual authentication. The supplicant and the authentication server both require a certificate. This is a more labor-intensive installation since the certificates need to be deployed on all devices that will authenticate to the server. Due to the requirement for client certificates, a working PKI (public key infrastructure) is needed to deploy EAP-TLS. Authentication is performed over a TLS tunnel using the client and server certificates.

Figure 12 - EAP-TLS



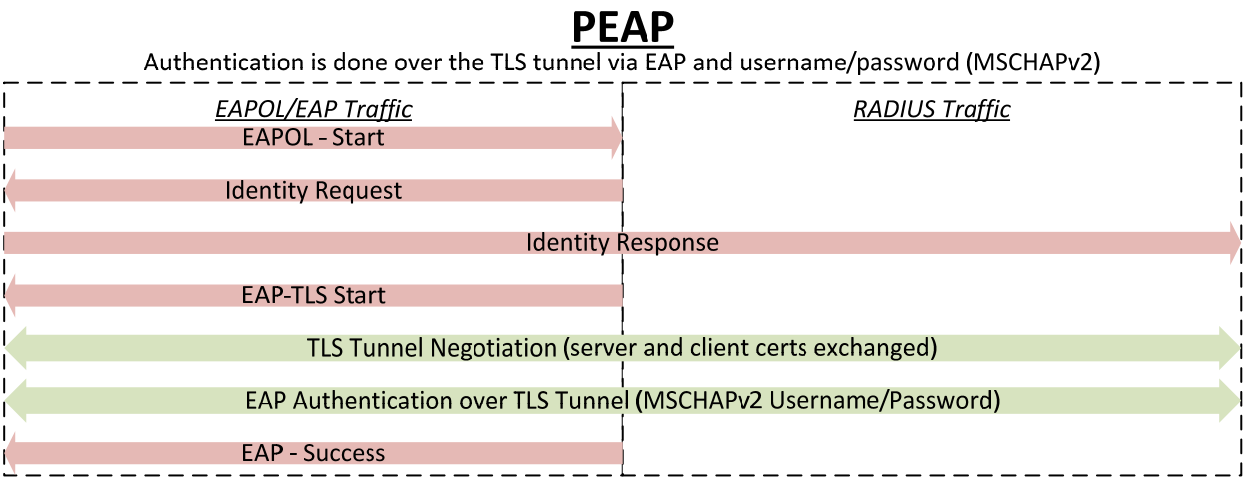
EAP-TLS RFCS

- RFC 2716 (original)
- RFC 5216 (updated)

EAP-PEAP

EAP-PEAP (EAP- Protected Extensible Authentication Protocol), also known as **Protected EAP**, is a widely supported EAP method. It is similar to EAP-TLS but does not have the requirement for client certificates. Authentication is performed over a TLS tunnel using a username and password with MSCHAPv2. It provides good security without the needed PKI infrastructure.

Figure 13 - EAP-PEAP



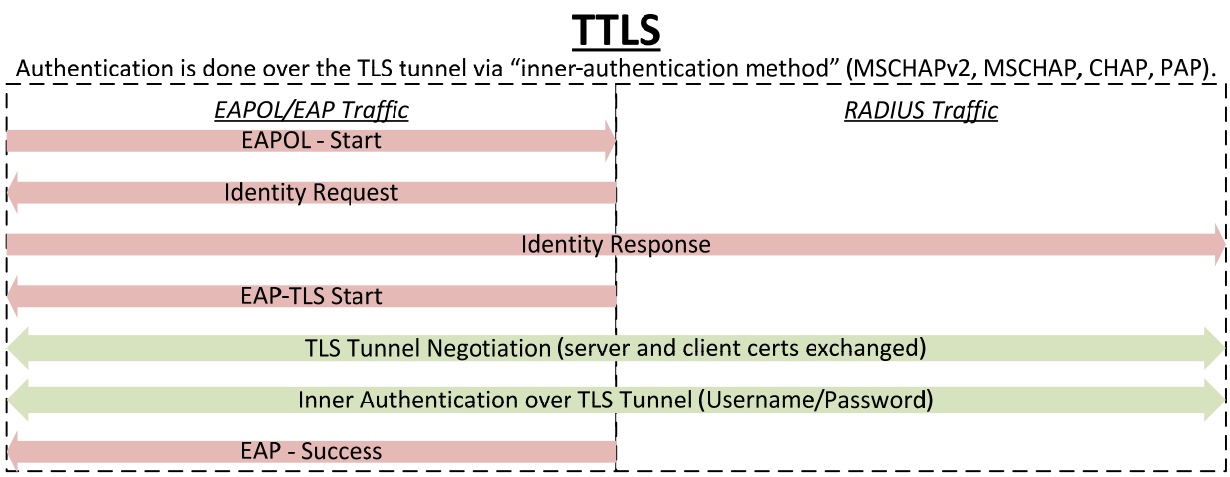
EAP-PEAP RFCS

- RFC 4017

EAP-TTLS

EAP-TTLS (EAP-Tunneled Transport Layer Security) performs authentication over the TLS tunnel using an "inner-authentication method". These methods include MSCHAPv2, MSCHAP, CHAP, and PAP. The inner-authentication methods are protected by the TLS tunnel. The TLS tunnel does not require a client certificate.

Figure 14 - EAP-TTLS



EAP-TTLS RFCS

- RFC 5281

EAP-FAST

EAP-FAST (EAP-Flexible Authentication via Secure Tunneling) was developed by Cisco to replace the older, insecure LEAP protocol. EAP-FAST is generally used when the authentication server is a Cisco ACS device. Authentication is performed over a TLS tunnel created using a PAC (Protected Access Credentials) as opposed to a client certificate. An internal protocol such as MSCHAPv2 is used within the TLS tunnel.

A PAC is a secret key file that is unique to each user. The PACs are generated by the authentication server, in this case a Cisco ACS, using a master key that only the ACS server knows. The PACs are used instead of certificates, which are used in other EAP methods, to create the secure tunnel for authentication to use. This removes the certificate management needed for other methods.

EAP-FAST has three phases:

PHASE 0

Phase 0 is used to provide an end user, such as an MFP, with a PAC. A Diffie-Hellman key exchange is used to create a secure tunnel between the client and the server. This tunnel is used in the provisioning of the PAC. MS-CAHPv2 authentication is used within this tunnel. If authentication is successful, the PAC is provided.

PAC provisioning can be anonymous or authenticated. TA Triumph-Adler/UTAX devices only support authenticated PAC provisioning. This means that the MFP must provide valid credentials to receive a PAC. Anonymous, which is not supported, would allow the PAC to be provisioned without prior authentication.

Even when Phase 0 is successful, it does not provide network access. Phase 0 is only for PAC provisioning.

Note: Phase 0 is optional, as PACs could be premade and provisioned manually. However, manual PAC provisioning is not supported by TA Triumph-Adler/UTAX devices. **Only authenticated PAC provisioning is supported**.

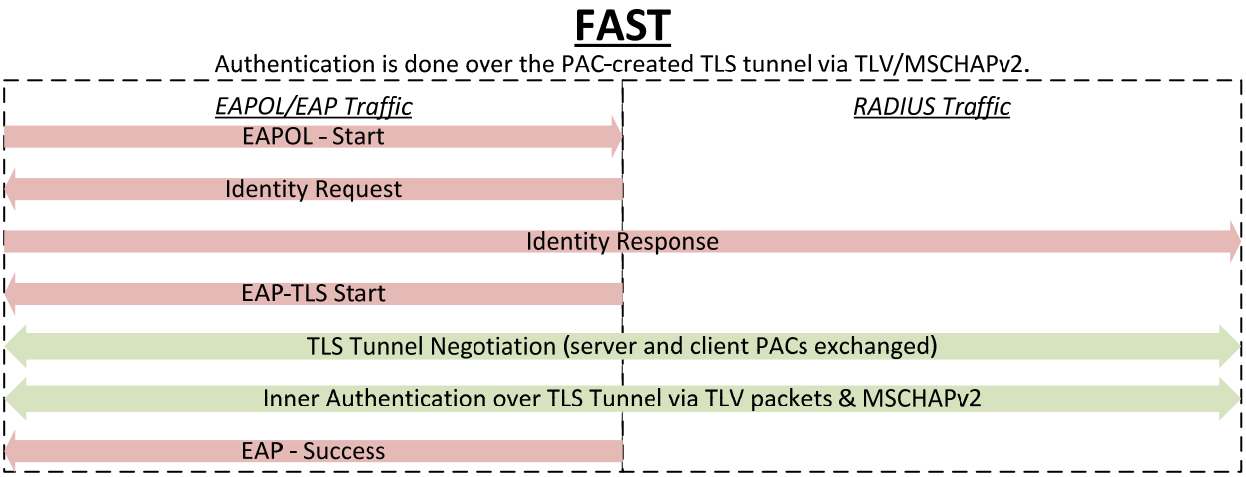
PHASE 1

The client and the server establish a TLS tunnel using the PAC provisioned in Phase 0.

PHASE 2

The authentication server authenticates the client credentials. If successful, the server will allow or deny network access.

Figure 15 - EAP-FAST



EAP-FAST RFCS

- RFC 4851