CHAPTER 1

# Overview of EAP-FAST

This chapter provides an overview of EAP-FAST (Flexible Authentication via Secure Tunneling). This chapter includes the following sections:



## Introduction to EAP-FAST

**Note** For additional information about EAP-FAST, see RFC4851.

EAP-FAST is an EAP method that enables secure communication between a client and an authentication server by using Transport Layer Security (TLS) to establish a mutually authenticated tunnel. Within the tunnel, data in the form of type, length, and value (TLV) objects are used to send further authentication-related data between the client and the authentication server.

EAP-FAST supports the TLS extension as defined in RFC 4507 to support the fast re-establishment of the secure tunnel without having to maintain per-session state on the server. EAP-FAST-based mechanisms are defined to provision the credentials for the TLS extension. These credentials are called Protected Access Credentials (PACs).

EAP-FAST provides the following:

- Mutual authentication

  An EAP server must be able to verify the identity and authenticity of the client, and the client must be able to verify the authenticity of the EAP server.

- Immunity to passive dictionary attacks

  Many authentication protocols require a password to be explicitly provided (either as cleartext or hashed) by the client to the EAP server. The communication of the weak credential (such as a password) must be immune from eavesdropping.

- Immunity to man-in-the-middle (MitM) attacks

  In establishing a mutually authenticated protected tunnel, the protocol must prevent adversaries from successfully interjecting information into the communication between the client and the EAP server.

- Flexibility to enable support for most password authentication interfaces

Many different password interfaces exist to authenticate a client—for example, Microsoft Challenge Handshake Authentication Protocol (MS-CHAP), Lightweight Directory Access Protocol (LDAP), and One-Time Password (OTP). EAP-FAST provides support for these different password types.

- Efficiency in computational and power resources

  Especially when using wireless media, clients have limited computational and power resources. EAP-FAST enables network access communication to occur in a more efficient manner.

- Flexibility to extend the communications inside the tunnel

  Because network infrastructures are becoming increasingly complex, authentication, authorization, and accounting is also becoming more complex. For example, there are instances in which multiple existing authentication protocols are required to achieve mutual authentication. Also, different protected conversations might be required to achieve the proper authorization when a client has successfully authenticated.

- Minimize authentication server requirements for per-user authentication

  With large deployments, it is typical to have several servers that act as authentication servers for several clients. A client uses the same shared secret to secure a tunnel in much the same way that is uses a username and password to gain access to the network. EAP-FAST facilitates the use of a single strong shared secret by the client, while enabling the authentication servers to minimize the per-user and device state that they must cache and manage.

# How EAP-FAST Works

The following sections describe how EAP-FAST works:



## Two-Phase Tunneled Authentication

EAP-FAST uses a two-phase tunneled authentication process.

In the first phase of authentication, EAP-FAST employs the TLS handshake to provide an authenticated key exchange and to establish a protected tunnel between the client and the authentication server. The tunnel protects client identity information from disclosure outside the tunnel. During this phase, the client and the server engage in EAP-FAST version negotiation to ensure that they are using a compatible version of the protocol.

After the tunnel is established, the second phase of authentication begins. The client and server communicate further to establish the required authentication and authorization policies. This phase consists of a series of requests and responses that are encapsulated in TLV objects. The TLV exchange includes the EAP method to be used within the protected tunnel. For more information about TLV objects and format, see section 4.2 of RFC 4851.

The EAP-FAST module offers a variety of EAP-FAST configuration options, including whether automatic or manual PAC provisioning is used to establish a tunnel, whether or not server certificate is used to establish a tunnel, what type of user credentials to use for authentication and provisioning, and what type of authentication method to use to in the established tunnel.

# Protected Access Credentials

Protected Access Credentials (PACs) are credentials that are distributed to clients for optimized network authentication. PACs can be used to establish an authentication tunnel between the client and the authentication server (the first phase of authentication as described in the "Two-Phase Tunneled Authentication" section on page 1-2). A PAC consists of, at most, three components: a shared secret, an opaque element, and other information.

The shared secret component contains the pre-shared key between the client and authentication server. Called the PAC-Key, this pre-shared key establishes the tunnel in the first phase of authentication.

The opaque component is provided to the client and is presented to the authentication server when the client wants to obtain access to network resources. Called the PAC-Opaque, this component is a variable length field that is sent to the authentication server during tunnel establishment. The EAP server interprets the PAC-Opaque to obtain the required information to validate the client's identity and authentication. The PAC-Opaque includes the PAC-Key and may contain the PAC's client identity.

The PAC might contain other information. Called PAC-Info, this component is a variable length field that is used to provide, at a minimum, the authority identity of the PAC issuer (the server that created the PAC). Other useful but not mandatory information, such as the PAC-Key lifetime, can also be conveyed by the PAC-issuing server to the client during PAC provisioning or refreshment.

PACs are created and issued by a PAC authority, such as Cisco Secure ACS, and are identified by an ID. A user obtains his or her own copy of a PAC from a server, and the ID links the PAC to a profile.

Persistent PACs, such as machine PACs, are stored in the EAP-FAST registry and encrypted. These PACs are also protected with access control lists (ACLs) so only designated users (the owners of the PACs) and members of privileged user groups (for example, administrators) can access them. Machine PACs are stored globally so that all users of a machine can use the PACs.

All PACs are encrypted and tied to the host machine with Microsoft Crypto API (CryptoProtectData). PACs cannot be copied and used on other machines.

All non-persistent PACs, such as User Authorization PACs, are stored in volatile memory and do not persist after reboot or after a user has logged off.

# Server Certificate Validation

As a part of TLS negotiation in the first phase of EAP-FAST authentication, the authentication server presents the client with a certificate. The client must verify the validity of the EAP server certificate and also examines the EAP server name that is presented in order to determine if the server can be trusted.