About the 802.1x Protocol

Wireless Networking

Protocol operation

Port entities

802.1X-2001 defines two logical port entities for an authenticated port the "controlled port" and the "uncontrolled port". The controlled port is manipulated by the 802.1X PAE (Port Access Entity) to allow (in the authorized state) or prevent (in the unauthorized state) network traffic ingressing and egressing to/from the controlled port. The uncontrolled port is used by the 802.1X PAE to transmit and receive EAPOL frames.

802.1X-2004 defines the equivalent port entities for the supplicant; so a supplicant implementing 802.1X-2004 may prevent higher level protocols being used if it is not content that authentication has successfully completed. This is particularly useful when an EAP method providing Mutual Authentication [1] is used, as the supplicant can prevent data leakage when connected to an unauthorized network.

Typical authentication progression

- **Initialization** On detection of a new supplicant, the port on the switch (authenticator) is enabled and set to the "unauthorized" state. In this state, only 802.1X traffic is allowed; other traffic, such as DHCP [2] and HTTP [3], is dropped.
- **Initiation** To initiate authentication the authenticator will periodically transmit EAP-Request Identity frames to a special Layer 2 address on the local network segment. The supplicant listens on this address, and on receipt of the EAP-Request Identity frame it responds with an EAP-Response Identity frame containing an identifier for the supplicant such as a User ID. The authenticator then encapsulates this Identity response in a RADIUS Access-Request packet and forwards it on to the authentication server. The supplicant may also initiate or restart authentication by sending an EAPOL-Start frame to the authenticator, which will then reply with an EAP-Request Identity frame.
- **Negotiation** *(Technically EAP negotiation)* The authentication server sends a reply (encapsulated in a RADIUS Access-Challenge packet) to the authenticator, containing an EAP Request specifying the EAP Method (The type of EAP based authentication it wishes the supplicant to perform). The authenticator encapsulates the EAP Request in an EAPOL frame and transmits it to the supplicant. At this point the supplicant can NAK the requested EAP Method and respond with the EAP Methods it's willing to perform, or start the requested EAP Method.
- **Authentication** If the authentication server and supplicant agree on an EAP Method, EAP Requests and Responses are sent between the supplicant and the authentication server (translated by the authenticator) until the authentication server responds with either an EAP-Success message (encapsulated in a RADIUS Access-Accept packet), or an EAP-Failure message (encapsulated in a RADIUS Access-Reject packet). If authentication is successful, the authenticator sets the port to the "authorized" state and normal traffic is allowed, if it is unsuccessful the port remains in the "unauthorized" state. When the supplicant logs off, it sends an EAPOL-logoff message to the authenticator, the authenticator then sets the port to the "unauthorized" state, once again blocking all non-EAP traffic

Source URL: http://uni.edu/its/support/article/591

Links:
[1] http://en.wikipedia.org/wiki/Mutual_authentication
[2] http://en.wikipedia.org/wiki/DHCP
[3] http://en.wikipedia.org/wiki/HTTP