Authentication, Authorization, Accounting (AAA)

Raj Jain
Washington University in Saint Louis
Saint Louis, MO 63130
Jain@cse.wustl.edu

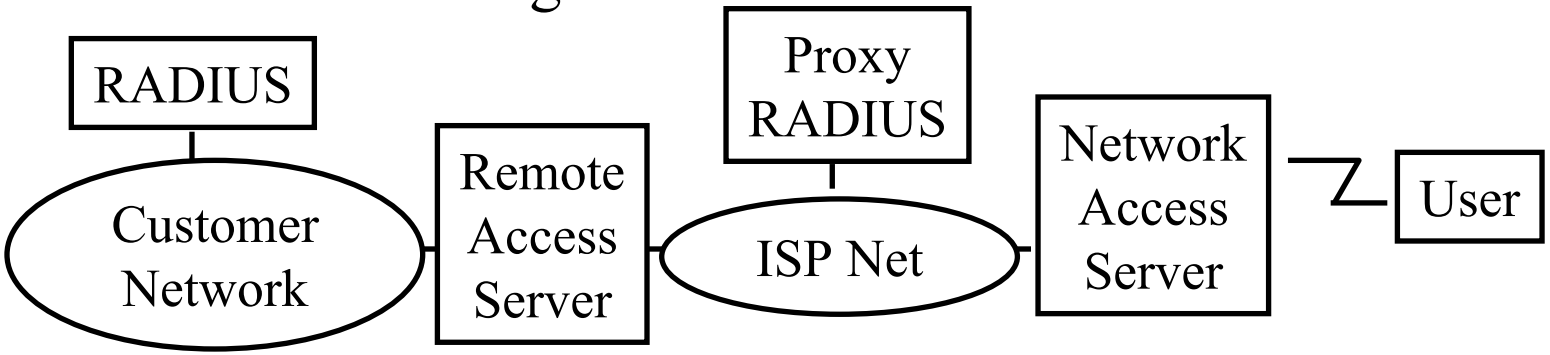
Audio/Video recordings of this lecture are available at:

http://www.cse.wustl.edu/~jain/cse571-11/

- RADIUS
- Authentication Protocols: PAP, CHAP, MS-CHAP
- Extensible Authentication Protocol (EAP)
- EAP Upper Layer Protocols
- 802.1X

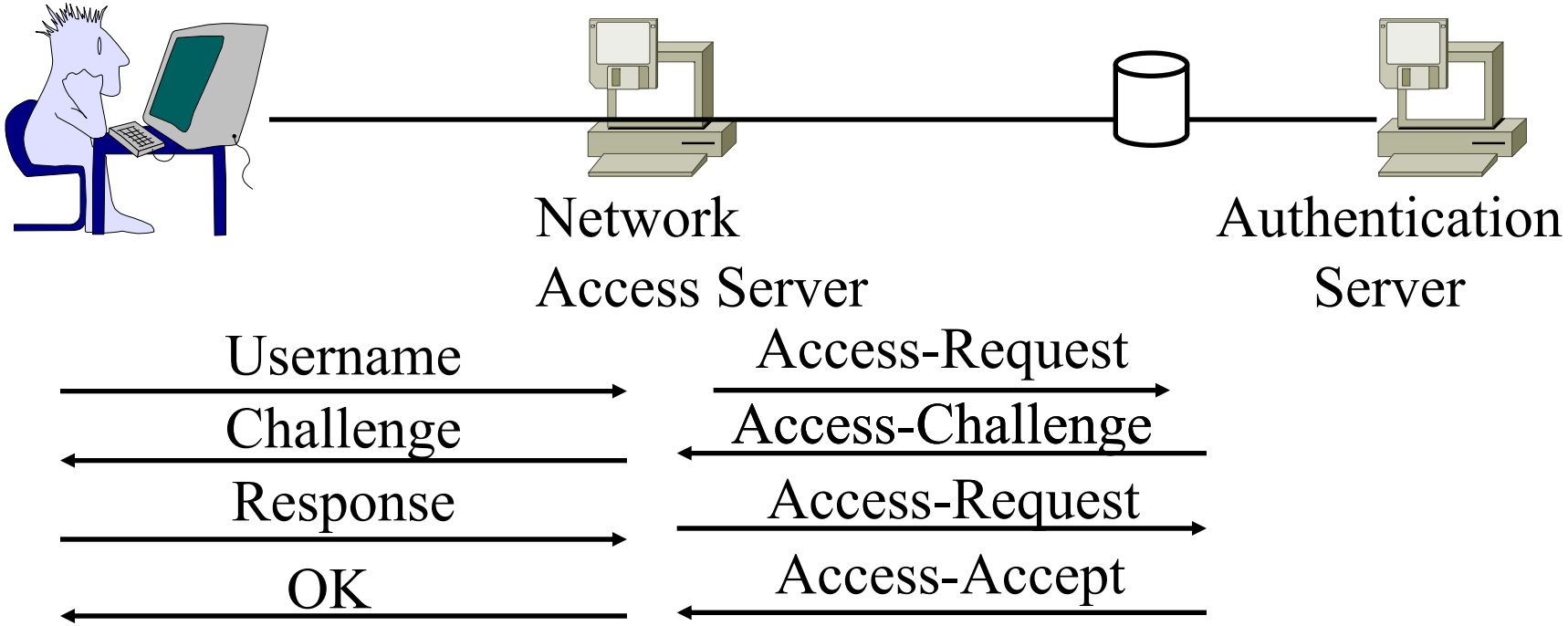
RADIUS

- Remote Authentication Dial-In User Service
- Central point for Authorization, Accounting, and Auditing data ⇒ **AAA** server
- Network Access servers get authentication info from RADIUS servers
- Allows RADIUS Proxy Servers ⇒ ISP roaming alliances
- Uses UDP: In case of server failure, the request must be re-sent to backup ⇒ Application level retransmission required
  - TCP takes to long to indicate failure



- Ref: http://en.wikipedia.org/wiki/RADIUS

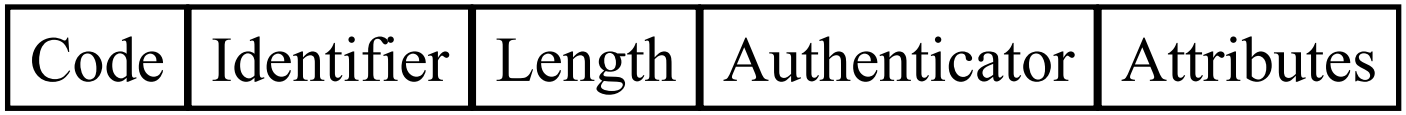
RADIUS Messages



- Four Core Messages: Request, Challenge, Accept, Reject.
- Message Format: Code is the message type.
  Identifier is used to match request/response.

Code	Identifier	Length	Authenticator	Attributes

RADIUS Packet Format



Codes:

1 = Access Request

2 = Access Accept

3 = Access Reject

4 = Accounting request

5 = Accounting Response

11 = Access Challenge

12 = Server Status (experimental)

13 = Client Status (Experimental)

255 = Reserved

RADIUS Accounting

- RFC 2866, June 2000
- Client sends to the server:
  - Accounting Start Packet at service beginning
  - Accounting Stop Packet at end
- All packets are acked by the server
- Packet format same as in authentication

Problems with RADIUS

- Does not define standard failover mechanism
  $\Rightarrow$ varying implementations
- Original RADIUS defines integrity only for response packets
- RADIUS extensions define integrity for EAP sessions
- Does not support per-packet confidentiality
- Billing replay protection is assumed in server.
  Not provided by protocol.
- IPsec is optional
- Runs on UDP $\Rightarrow$ Reliability varies between implementation.
  Billing packet loss may result in revenue loss.
- RADIUS does not define expected behavior for proxies, redirects, and relays $\Rightarrow$ No standard for proxy chaining

Problems with RADIUS (Cont)

- Does not allow server initiated messages
  ⇒ No On-demand authentication and unsolicited disconnect
- Does not define data object security mechanism
  ⇒ Untrusted proxies can modify attributes
- Does not support error messages
- Does not support capability negotiation
- No mandatory/non-mandatory flag for attributes
- Servers name/address should be manually configured in clients ⇒ Administrative burden
  ⇒ Temptation to reuse shared secrets

Diameter Base Protocol

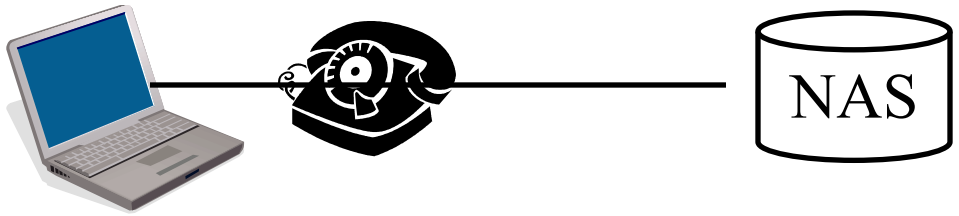
- Enhanced RADIUS. Light weight.
- Can use UDP, TCP, SCTP (Stream Control Transmission Protocol)
- PDU format incompatible with RADIUS
- Can co-exist with RADIUS in the same network
- Defines standard failover algorithm
- Supports:
  - Delivery of attribute-value pairs (AVPs)
  - Capability negotiation
  - Error notification
  - Ability to add new commands and AVPs
  - Discovery of servers via DNS
  - Dynamic session key derivation via TLS

Ref: http://en.wikipedia.org/wiki/DIAMETER

Diameter Base Protocol (Cont)

- All data is delivered in the form of AVPs
- AVPs have mandatory/non-mandatory bit
- Support for vendor specific Attribute-Value-Pairs (AVPs) and commands
- Authentication and privacy for policy messages
- Peer-to-peer protocol ⇒ any node can initiate request.
- Servers can send unsolicited messages to Clients
  ⇒ Increases the set of applications
- Documents: Base, transport profile, applications
- Applications: NAS, Mobile IP, Credit control (pre-paid, post-paid, credit-debit), 3G, EAP, SIP

Password Authentication Protocol (PAP)



- RFC 1334, Oct 1992
- Authenticator sends a authentication request
- Peer responds with a username and password in plain text
- Authenticator sends a success or failure
- Code: 1=Auth Request, 2=Auth Ack, 3=Auth Nak

Code	ID	Len	Name Len	Name Val	Pswd Len	Pswd Val
1B	1B	2B	1B	Var	1B	Var

Code	ID	Len	Success/Failure Message
1B	1B	2B	1B

Ref: http://en.wikipedia.org/wiki/Password_Authentication_Protocol

CHAP

- Challenge Handshake Authentication Protocol
- RFC 1994, August 1996
- Uses a shared secret (password)
- Authenticator sends a challenge
- Peer responds with a MD5 checksum hash of the challenge
- Authenticator also calculates the hash and sends success or failure
- Requires both ends to know the password in plain text
- Replay attack prevention ⇒ Use a different challenge every time

Ref: http://en.wikipedia.org/wiki/Challenge-handshake_authentication_protocol

MS-CHAP

- Microsoft version of CHAP
- MS-CHAP in RFC 2433, Oct 1998
- Does not require password in plain text
- Uses hash of the password
- 8B challenge ⇒ 24B LM (LAN Manager) compatible response, 24B NTLM compatible response and 1B NTLM flag
- LM passwords are limited to 14 case-insensitive OEM characters
- NT passwords are 0 to 256 case-sensitive Unicode characters
- Flag ⇒ NT response is meaningful and should be used
- Also allows users to change password
- MS-CHAPv2 in Windows 2000 onwards.

Ref: http://en.wikipedia.org/wiki/MS-CHAP

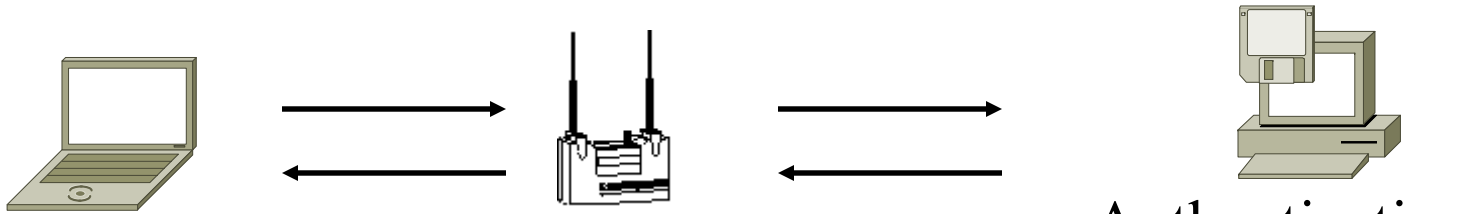
Extensible Authentication Protocol (EAP)

- Each authentication protocols required a new protocol
  ⇒ Extensible Authentication Protocol
- Initially developed for point-to-point protocol (PPP)
- Allows using many different authentication methods
- Single-Step Protocol ⇒ Only one packet in flight
  ⇒ Duplicate Elimination and retransmission
  Ack/Nak ⇒ Can run over lossy link
- No fragmentation. Individual authentication methods can deal with fragmentation. One frag/round trip ⇒ Many round trips
- Allows using a backend authentication server ⇒ Authenticator does not have to know all the authentication methods
- Can run on any link layer (PPP, 802, ...). Does not require IP.
- RFC 3748, "EAP," June 2004.

Ref: http://en.wikipedia.org/wiki/Extensible_Authentication_Protocol

EAP Terminology

- Peer: Entity to be authenticated = Supplicant
- Authenticator: Authenticating entity at network boundary
- Authentication Server: Has authentication database
- EAP server = Authenticator if there is no backend Authentication Server otherwise authentication server
- Master Session Key (MSK)= Keying material agreed by the peer and the EAP server. At least 64B. Generally given by the server to authenticator.

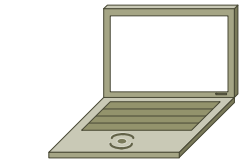


EAP Exchange

- EAP Message Format:

Code	Identifier	Length	Data
8b	8b	16b	

- Only four types of messages:



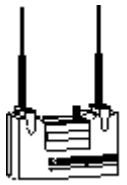
Supplicant

Request (01)

Response (02)

Success (03)

Failure (04)

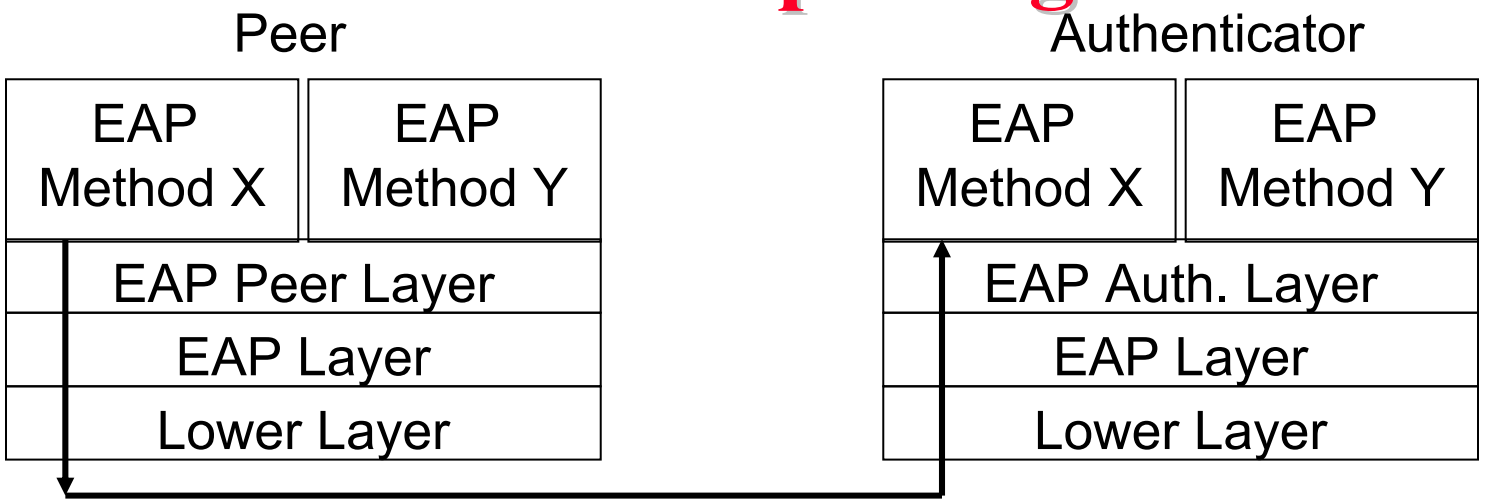


Authenticator

- Identifier is incremented for each message.
  Identifier in response is set equal to that in request.
- Type field in the request/response indicates the authentication.
  Assigned by Internet Assigned Number Authority (IANA)

Code	Identifier	Length	**Type**	Data

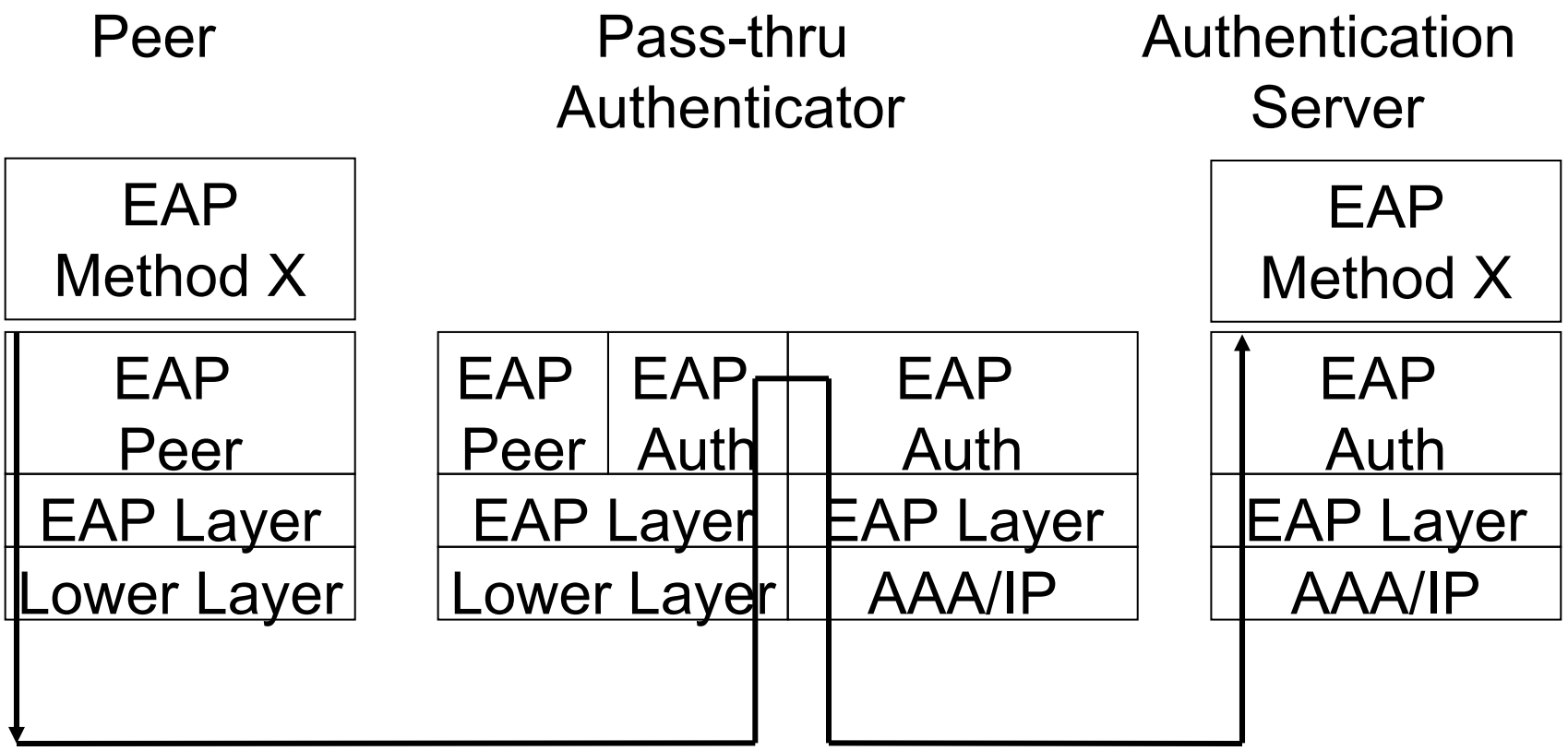
EAP Multiplexing Model



- EAP Layer demultiplexes using code. Code 1 (request), 3 (success), and 4 (failure) are delivered to the peer layer
- Code 2 (response) is delivered to the EAP authenticator layer.
- Both ends may need to implement peer layer and authenticator layer for mutual authentication
- Lower layer may be unreliable but it must provide error detection (CRC)
- Lower layer should provide MTU of 1020B or greater

Ref: RFC 3748

EAP Pass through Authenticator



- EAP Peer/Auth layers demultiplex using "type" field.

EAP Upper Layer Protocols

- Lightweight EAP (LEAP): Uses MS-CHAP. Not secure.
- EAP-TLS: Transport Level Security. Both sides need certificates
- EAP-TTLS: Tunneled TLS. Only server certificates. Secure tunnel for peer.
- EAP-FAST: Flexible Authentication via Secure Tunneling. Certificates optional. Protected tunnels.
- Protected EAP (PEAP): Server Certificates. Client password.
- PEAPv1 or EAP-GTC: Generic Token Cards. Client uses secure tokens.
- EAP-SIM: Subscriber Identity Module used in GSM. 64b keys.
- EAP-AKA: Authentication and Key Agreement. Used in 3G. 128b keys.
- EAP-PSK: Pre-shared key+AES-128 to generate keys
- EAP-IKEv2: Internet Key Exchange. Mutual authentication. Certificate, Password, or Shared secret

Ref: http://en.wikipedia.org/wiki/Protected_Extensible_Authentication_Protocol

Security Token

- Security Token = Small hardware device carried by users. May store cryptographic keys, biometric data (finger print), PIN entry pad.
- Based on USB, Bluetooth, Cell phones (SMS or Java)
- Use smart cards
- Two-factor authentication = What you have and what you know

[Wikipedia]

Ref: http://en.wikipedia.org/wiki/Security_token

One-Time Password

- Three Types:
  1. Use a math algorithm to generate a new password based on previous
  2. Uses time to generate password
     ⇒ Synchronized time between server and client
  3. Use a math algorithm to generate a new password based on a challenge from the server and a counter.
- Time synchronized approach allows users to generate password and not use it. The server may compare with the next n passwords to allow for time miss-synchronization.
- Non-time synchronized OTP do not need to be powered all the time ⇒ battery lasts long. Have been attacked by phishing. Time-based OTP need to be used right-away.

Ref: http://en.wikipedia.org/wiki/One-time_password

EAP over LAN (EAPOL)

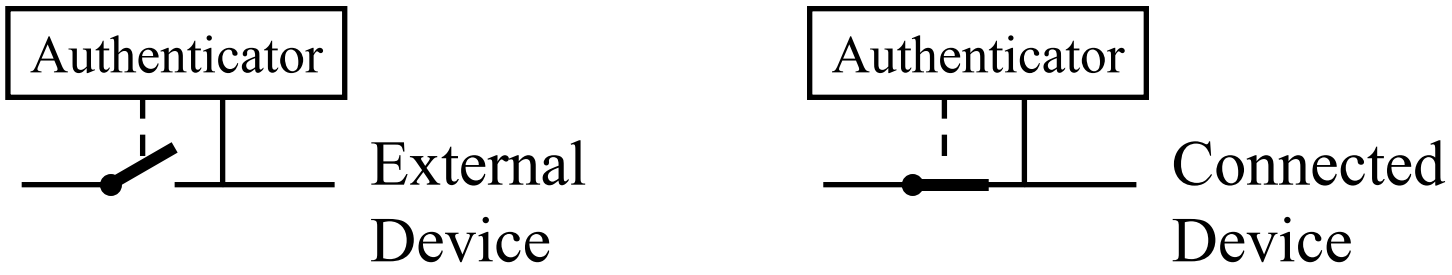
- EAP was designed for Point-to-point line
- IEEE extended it for LANs ⇒ Defines EAPOL
- Added a few more messages and fields
- Five types of EAPOL messages:
  - EAPOL Start: Sent to a multicast address
  - EAPOL Key: Contains encryption and other keys sent by the authenticator to supplicant
  - EAPOL packet: Contains EAP message (Request, Response, Success, Failure)
  - EAPOL Logoff: Disconnect
  - EAPOL Encapsulated-ASF-Alert: Management alert
- Message Format: Version=1, Type=start, key, …,

Ethernet Header	Version	Type	Packet Body Len	Packet Body

Ref: http://en.wikipedia.org/wiki/Eapol

802.1X

- Authentication *framework* for IEEE802 networks
- Supplicant (Client), Authenticator (Access point), Authentication server
- No per packet overhead ⇒ Can run at any speed
- Need to upgrade only driver on NIC and firmware on switches
- User is not allowed to send any data until authenticated



Ref: http://en.wikipedia.org/wiki/802.1x

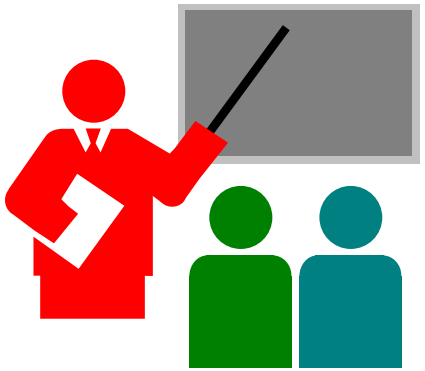
802.1X Authentication

Station | **Access Point** | **Authentication Server**

Can I connect please?	Associate →	
What's your user name?	EAP Identity Request ←	
My user name is john	EAP Identity Response →	EAP Identity Response →
What's your password?	EAP Auth Request ←	EAP Auth Request ←
My password is mary?	EAP Auth Response →	EAP Auth Response →
You can connect!	EAP-Success ←	EAP-Success ←

- Authentication method can be changed without upgrading switches and access points
- Only the client and authentication server need to implement the authentication method

Summary



- RADIUS allows centralized authentication server and allows roaming
- EAP allows many different authentication methods to use a common framework ⇒ Authenticators do not need to know about authentication methods
- Many variations of EAP authentication methods depending upon certificates, shared secrets, passwords
- 802.1X adds authentication to LAN and uses EAPOL

Homework 23

- How would you implement Kerberos v4 over EAP in a LAN environment. Show the sequence of EAP messages that will be sent for authentication and key generation. Show also EAPOL headers on the messages.
- Hint: Use the 6 messages used in Kerberos and put EAPOL headers on them.