

AAA

Part II

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/cse571-07/>



- ❑ TACACS, TACACS+
- ❑ RADIUS, Packet Format, Accounting
- ❑ Problems with RADIUS
- ❑ Diameter Base Protocol
- ❑ AAA Transport Profile
- ❑ AAA Key Management Principles

TACACS

- ❑ Terminal Access Controller Access-Control System
- ❑ Routing nodes in ARPANet were called IMPs.
- ❑ IMPs with dial up access were called TIPs.
- ❑ BBN developed TACACS for ARPANET
- ❑ AAA server is a process in a UNIX server - called TACACS daemon.
- ❑ Uses UDP port 49
- ❑ Username and passwords were sent in clear for authentication
⇒ No longer used
- ❑ Cisco adopted TACACS for terminal servers extended TACACS or XTACACS

TACACS+

- ❑ Terminal Access Controller Access-Control System Plus
- ❑ Cisco's further improved version of TACACS and XTACACS
- ❑ Not compatible with TACACS
- ❑ Payload is encrypted
- ❑ Described in draft-grant-tacacs-02.txt, Jan 1997.
- ❑ Uses TCP port 49

RADIUS

- ❑ RFC 2138, June 2000
- ❑ UDP port 1812
- ❑ Why UDP?
 - In case of server failure, the request must be re-sent to backup ⇒ Application level retransmission required
 - TCP takes too long to indicate failure
 - Stateless protocol

RADIUS Packet Format

Code	Identifier	Length	Authenticator	Attributes
1B	1B	2B	16B	

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 Packet Format (Cont)

- ❑ 16B Authenticator is used to authenticate the reply from the RADIUS server
- ❑ In Access-request packets 16B random number is send as authenticator
- ❑ Password in packet
= MD5(Shared secret | authenticator) \oplus password
- ❑ Response Authenticator
= MD5(Code|ID|Length|Request Auth|Attributes|Shared secret)
- ❑ All attributes are TLV encoded.

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

RADIUS Server Implementations

Public domain software implementations:

- FreeRADIUS
- GNU RADIUS
- JRadius
- OpenRADIUS
- Cistron RADIUS
- BSDRadius
- TekRADIUS

Problems with RADIUS

- Does not define standard failover mechanism
⇒ 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 ⇒ Reliability varies between implementation.
Billing packet loss may result in revenue loss.
- RADIUS does not define expected behavior for proxies,
redirects, and relays ⇒ 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

- ❑ RFC 3588, Sep 2003
- ❑ Defines standard failover algorithm
- ❑ Runs over TCP and Stream Control Transmission Protocol (SCTP)
- ❑ PDU format incompatible with RADIUS
- ❑ Can co-exist with RADIUS in the same network
- ❑ 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

Diameter Base Protocol (Cont)

- ❑ All data is delivered in the form of AVPs
- ❑ AVPs have mandatory/non-mandatory bit
- ❑ Peer-to-peer protocol \Rightarrow any node can initiate request.
- ❑ Documents: Base, transport profile, applications
- ❑ Applications: NAS, Mobile IP, Credit control (pre-paid, post-paid, credit-debit), 3G, EAP, SIP

AAA Transport Profile

- ❑ RFC 3539, June 2003
- ❑ Network Access Identifier (NAI) = User ID
- ❑ Application driven vs. network driven:
Network is not the bottleneck for AAA messages
 \Rightarrow Application driven. No congestion issues.
- ❑ Slow Failover: TCP time outs \Rightarrow slow
- ❑ Use of Nagle Algorithm:
Many AAA messages are combined in one TCP message
- ❑ Multiple Connections:
Max 256 requests in progress between a client and a server
- ❑ Duplicate Detection: Servers and clients recognize duplicate request or responses and discard them.
 - A single request when duplicated can result in success and failure responses.

AAA Transport Profile (Cont)

- ❑ Invalidation of Transport Parameter Estimates: Timeouts should account for network congestion
- ❑ Inability to use fast re-transmit: most AAA protocols are always close to initial window set to 1 or 2
- ❑ Congestion Avoidance:
- ❑ Delayed Acks: application driven \Rightarrow explicit acks
- ❑ Premature failover: some implementation switch to backup server prematurely
- ❑ Head of line blocking: TCP queue may build up after a packet loss \Rightarrow hold up other AAA requests on the same connection
- ❑ Connection load balancing:

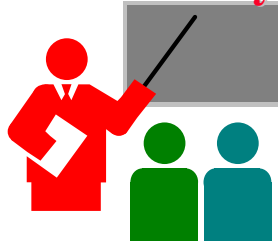
AAA Key Management Principles

- ❑ RFC 4962, July 2007 (Housley Criteria)
- ❑ Ability to negotiate crypto algorithms
 \Rightarrow Support multiple algorithm
- ❑ Ability to negotiate key derivation function is not required
- ❑ At least one suite of mandatory algorithms must be selected
- ❑ Use strong fresh session keys.
- ❑ Session keys must not be dependent on one another
 \Rightarrow Knowing a session key, Can't find another session key
 \Rightarrow Use nonce to ensure each session key is fresh.
- ❑ Include replay detection mechanism
- ❑ Authenticate all parties
- ❑ Lower layer identifiers used for authorization should be authenticated

AAA Key Management Principles (Cont)

- ❑ Both peer and authenticator must be authorized
⇒ Detect unauthorized authenticator
- ❑ Peer, Authenticator, Authentication server should have a common view of authorizations
- ❑ Cipher suite selection should be securely confirmed
⇒ detect roll-back attacks
- ❑ All keys should be uniquely named and key name should disclose key value
- ❑ Prevent domino effect ⇒ Compromise of a single entity must not compromise key material at other entities in other branches (may compromise children entities)
- ❑ Bind key to its context: use, who has access, life time. All entities with access to keying material should have the same context.

Summary



- ❑ TACACS and TACACS+ are legacy AAA protocols
- ❑ RADIUS provides good security but lacks sophisticated mechanisms required for failover
- ❑ Diameter is a replacement for RADIUS. Fixes most known shortcomings of RADIUS.