RADIUS + DTLS

http://www.ietf.org/internet-drafts/draft-dekok-radext-dtls-00.txt

Alan DeKok
FreeRADIUS

Introduction

- Crypto-agility is required
- *Forward* security is useful
  - We don't want to do this again
- RADIUS has ad-hoc security
  - authentication
  - encryption

Datagram TLS

- RFC 4347 was recently issued
- TLS over UDP (with some minor changes)
- Other WG's are using it
- OpenSSL supports it
  - Implementations of DTLS clients & servers exist

DTLS and Crypto-Agility

- TLS would appear to solve all crypto-agility requirements
  - Strong integrity checks
  - Strong encryption
  - Cryptographic negotiation
  - Designed by people who understand crypto
- Re-inventing crypto work is dangerous

Why DTLS isn't a good idea

- Heavy-weight (SSL)
- Relatively new
- additional implementation requirements
  - session state
  - connection oriented
  - Extra CPU / memory

Why DTLS is a good idea

- EAP already does TLS
- TLS is well tested and analyzed
- implementation requirements are minimal
  - request cache **is** session state
  - proxies already handle connections
- People already run RADIUS over IPSec...

Implementation

- http://crypto.stanford.edu/~nagendra/papers/dtls.pdf

```
int main(int argc, char **argv)
{
        s = socket(...);
        ...
        ...
        send(s, ...)
        ...
        recv(s, ...)
```

Implementation

- http://crypto.stanford.edu/~nagendra/papers/dtls.pdf

```
int main(int argc, char **argv)
{
        s = socket(...);
        SSL_init()
        ...
        SSL_write(s, ...)
        ...
        SSL_read(s, ...)
```

Benefits of DTLS

- Solves crypto-agility for once, and forever
- Maybe we don't need shared secrets any more?
- Connection oriented
  - Guaranteed delivery or notification
  - In-order delivery (point to point)

Diameter compatibility

- RADIUS + DTLS is a RADIUS transport layer change
- No changes to the RADIUS protocol
  - No messages, attributes, or enumerations
- Therefore no Diameter impact

RADIUS compatibility

- DTLS and RADIUS packets are orthogonal
  - non-standardized “resource allocation request”
  - less than one chance in 2^128 that packets can be confused
- RADIUS + DTLS can re-use the same ports
- Extending Id in DTLS sessions may be useful
  - increase number of in-flight packets

Discussion?

- DTLS is heavy-weight?
- Implementation details?
- Mandated crypto algorithms
- Tweaks to RADIUS to make DTLS more useful?
  - Session Id's are very limited