IETF 83

Analysis of NTP's Autokey Protocol

Dr. Dieter Sibold

Physikalisch-Technische Bundesanstalt

Stephen Röttger

Technische Universität Braunschweig

Motivation

PTB is Germany's National Metrology Institute (NMI)

Responsible for time dissemination (NTP and DCF77)

Authenticity is an increasing challenge for time dissemination via NTP

- Demand for securely authenticated time sources for home based smart meters; measuring of energy consumption and tariffing as a bases for billing
- Increasing number of requests for an authenticated (public) NTP time service

Issues with existing approaches

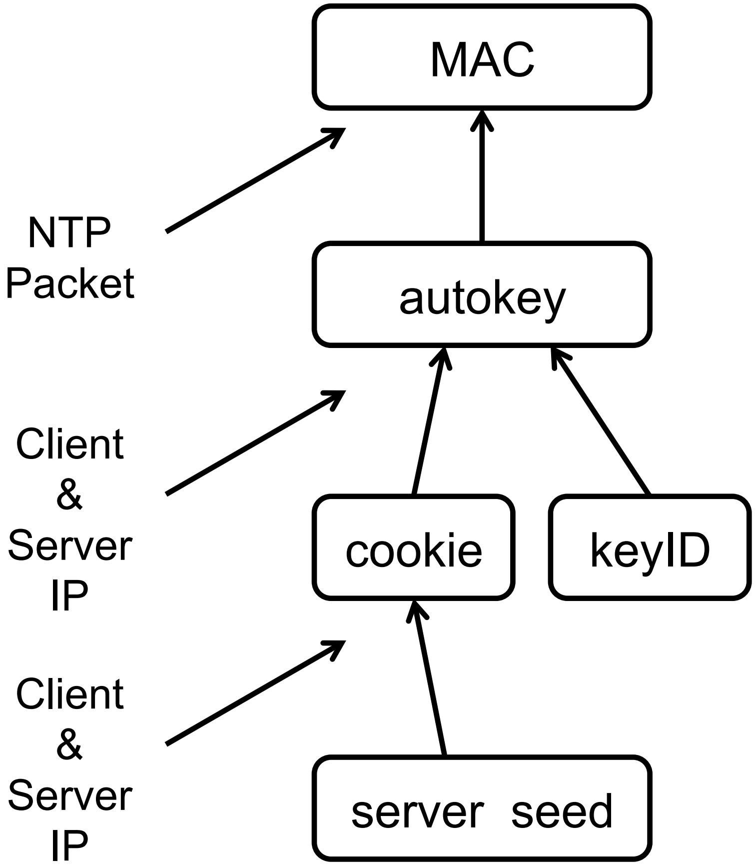
Pre-shared key

- Organizational effort
- No approval from official side (issues with compliance requirements)

Autokey

- Several vulnerabilities
  - in the Message Authentication Code (MAC) calculation and
  - the utilization of identity schemes
- Compatibility issues

Weak spots / MAC / Client-Server Mode



1. Server seed is only 32 bits long
   → Client can request a cookie and brute force the seed
2. The cookie is only 32 bits long; it is the only secret in the generation of the autokey (in Client-Server Mode)
   → An adversary can capture a packet and brute force the cookie
3. Client Identity Check: authenticity verification of the client is based on the client's IP address
   → An adversary can masquerade as the client and obtain the client's cookie encrypted with his own public key.

Weak spots / Identity Schemes

- **Trusted certification scheme provides no security enhancements**
- **Private certificate scheme works but requires pre-shared keys**
- **The three challenge response schemes (IFF, GQ, MV) are vulnerable against “man-in-the-middle” attacks**
- **The challenge response schemes are not applied adequately, which makes them non-effective**
  - → an adversary can send a response to a client challenge, which will be accepted by the client

Suggested autokey improvements

1. Augmentation of the bit length of the server seed and the cookie to 128 bits, respectively
2. Client authenticity check based on client's public key; cookie generation is then given by

   Cookie=Hash(public key of client || server seed)

3. Replacement of the identity schemes by a X.509 PKI
4. Optionally: signatures in extension fields cover the whole NTP packet
5. Optionally (for compliance reasons): utilization of NIST (or BSI) certified hash algorithms; e.g. key hashed MAC (HMAC)

Acknowledgement

Stephen Röttger

Technische Universität Braunschweig

Institute of Theoretical Information Technology

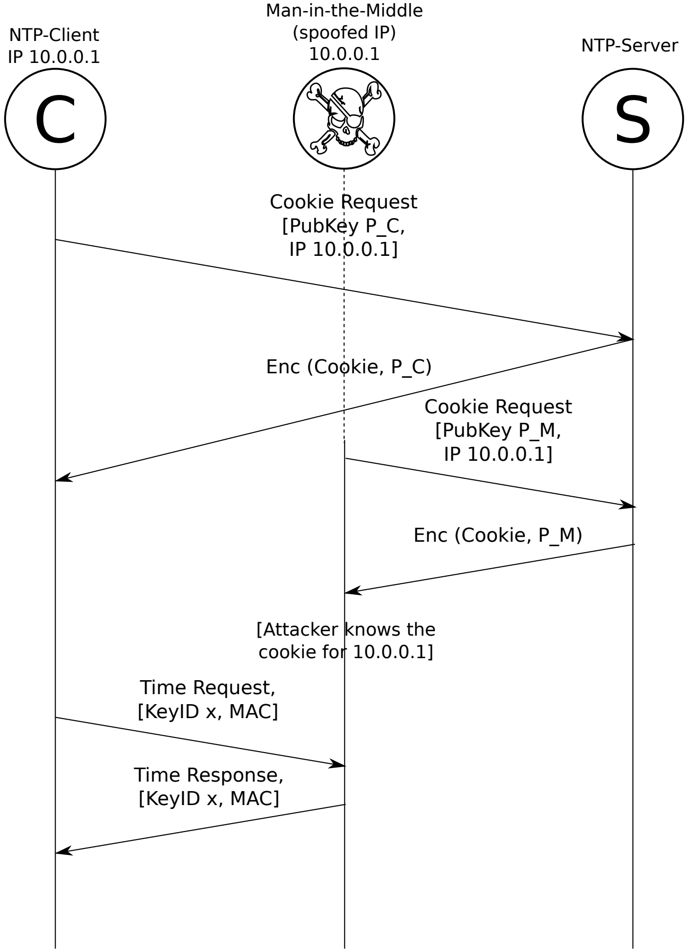
Generation of cookie, autokey and MAC

$$\text{Cookie} = \text{MSB}_{32}(H(\text{client IP}||\text{server IP}||0||\text{server seed}))$$

$$\text{autokey} = \text{H}(\text{server IP}||\text{client IP}||\text{keyID}||\text{cookie})$$

$$\text{MAC} = \text{H}(\text{autokey}||\text{NTP packet})$$

Exploit of the lacking identity check



Enc(Msg, P_X): Message 'Msg' encrypted with public key P_X