

SINO'2005

1ª Conferência Nacional sobre Segurança Informática nas Organizações

A Security Architecture for a Satellite Network Transport Architecture

André Zúquete (IT/IEETA/UA)

Ana Simões (SkySoft Portugal)



ieeta instituto de engenharia electrónica e telemática de aveiro



universidade de aveiro

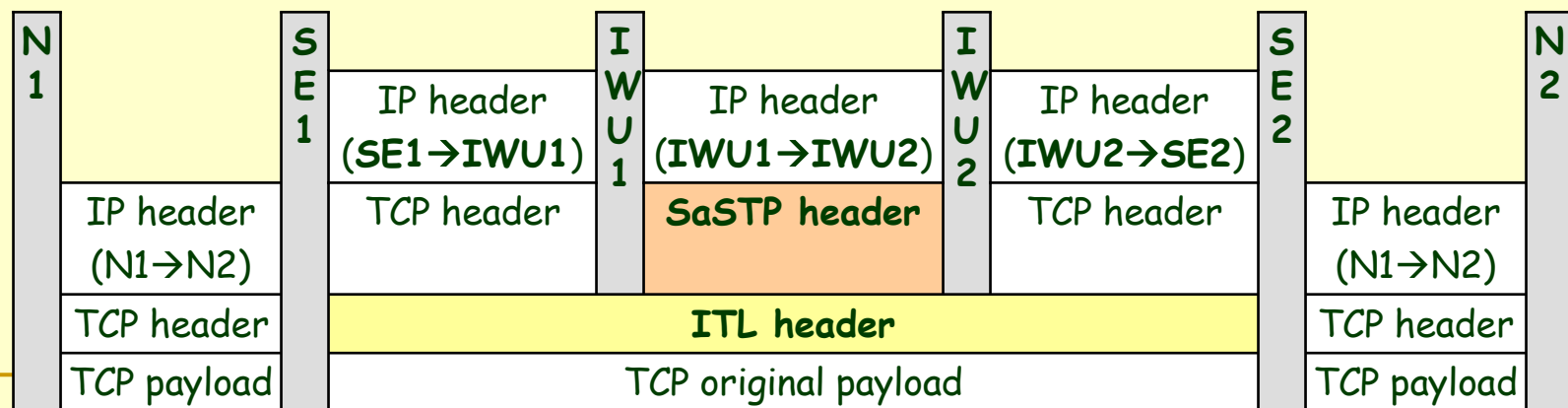
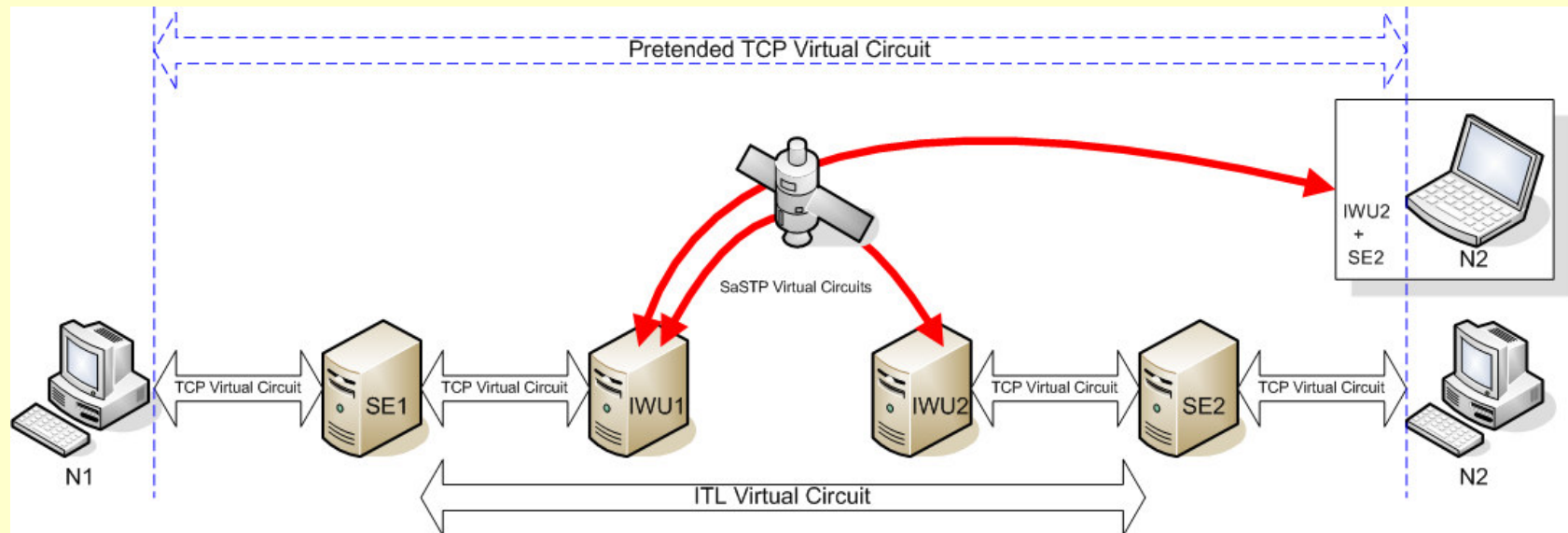


SaNTA (Satellite Network Transport Arch.):

Main goal and approach

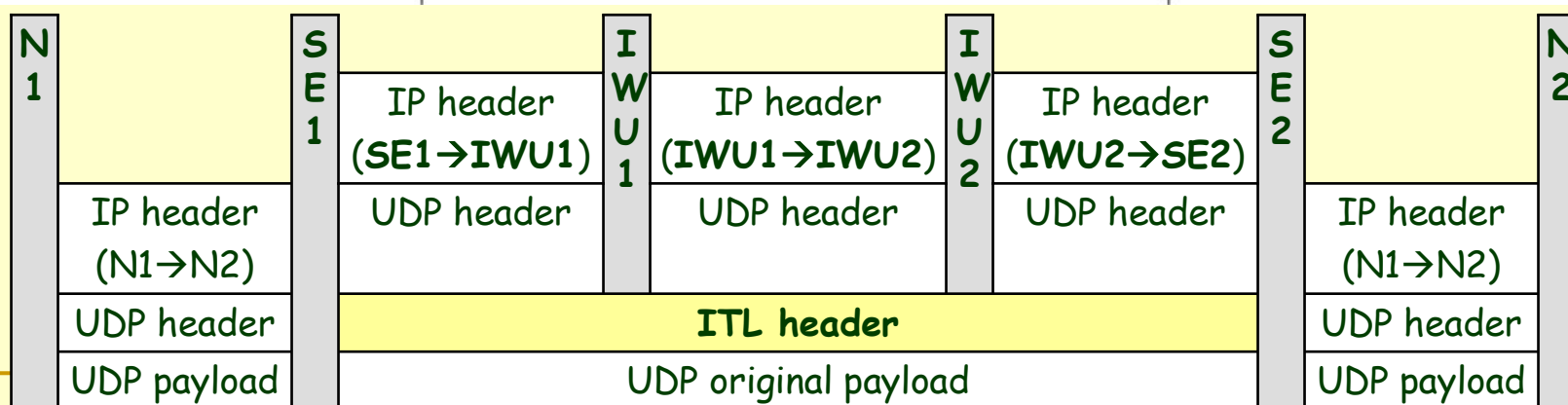
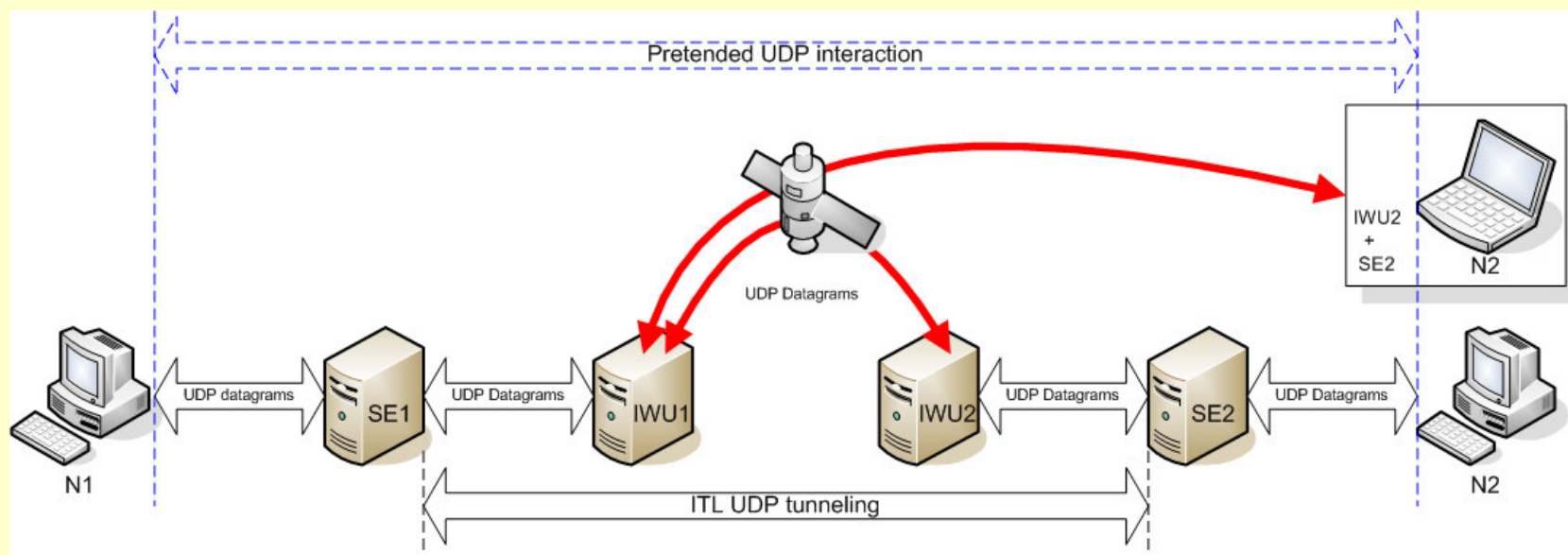
- Provide an efficient exploitation of TCP traffic over long-latency satellite links
 - The congestion control mechanism of TCP assumes that delays are a symptom of packet losses caused by congestion
- SaNTA breaks the traditional end-to-end paradigm
 - Uses a split architecture
 - Intermediate nodes impersonate end nodes for accelerating TCP acknowledges
 - Uses a new lightweight transport protocol
 - ITL (InterTransport Layer)
 - ITL accelerates TCP and tunnels other transport protocols (UDP)
 - Uses a new transport protocol for satellite links
 - SaSTP (SaNTA Satellite Transport Protocol)

SaNTA: TCP accelerating architecture



SaNTA:

UDP tunneling architecture



Security-related issues raised by SaNTA

- Most paradigms used in secure communication protocols enforce end-to-end security policies
 - Preventing intermediate nodes to take an active role in the communication
- Problems:
 - End-to-end confidentiality
 - Prevents SaNTA from accessing transport headers
 - Example: layer-2 VPNs (PPTP, L2TP), IPSec confidentiality
 - End-to-end peer authentication
 - Prevents SaNTA from impersonating end hosts/networks
 - Example: IPSec authentication of origin/destination hosts

SaNTA Security:

Goals

- Intrinsic security
 - Provide security solutions for protecting data flows handled by SaNTA entities
- Security tolerance
 - Provide the adequate mechanisms to tolerate end-to-end security mechanisms
- Architectural solution
 - Use state-of-art security mechanisms solutions for providing security

SaNTA security requirements:

Overview

■ Transparency

- Don't need to provide any security mechanisms directly to end-nodes and applications.
 - It's similar to an Internet core router
- It should not interfere with security requirements of end-nodes and applications

■ Secure added value

- It should not introduce new vulnerabilities in the traffic accelerated/tunneled

■ Correct exploitation

- It should be used only by authorized end-users or by authorized end-nodes/networks.
- It should not be abused by any Internet nodes

SaNTA security requirements:

Transparency (issues) (1/2)

- Host impersonation / payload inspection interfere with:
 - Layer-2 secure VPNs between end-hosts/networks (e.g. PPTP, L2TP)
 - Inspection impossible \Rightarrow no acceleration
 - Personification may be possible
 - End-to-end IPSec confidentiality (with ESP)
 - Inspection impossible \Rightarrow no acceleration
 - End-to-end IPSec authentication (with either AH or ESP)
 - Personification is possible but very hard (mainly with AH)

SaNTA security requirements:

Transparency (issues) (2/2)

- Host impersonation / payload inspection **don't interfere** with:
 - Security protocols above the transport layer (e.g. SSL, SSH, OpenVPN)
 - If they use TCP then they can be accelerated
 - TCP tunnels are globally accelerated (e.g. SSH tunnels)
 - If they use UDP they are tunneled
- Packet-filtering firewall & NAT boxes
 - ITL tunneling may help
 - Its easier to manage if SEs are within the protected perimeter
 - Closer to end-nodes/networks
 - Possibly within DMZs

SaNTA security requirements:

Transparency (overcoming problems)

- Two complementary approaches

- Mitigation

- Handle properly problematic traffics
 - Follow a best effort policy
 - Possible deployment: tunneling and relaying

- Security added value

- Deployment of internal security mechanisms
 - For providing a similar protection of data flows
 - They should allow end-users to securely abandon their end-to-end security mechanisms for benefiting from SaNTA acceleration

SaNTA security requirements:

Secure added value (1/2)

- SaNTA must be trusted by end-nodes/networks
 - Contractual relationship between the end-nodes and the company selling the services of a particular SaNTA infrastructure
- SaNTA must be authenticated by end-nodes/networks
 - End-nodes should authenticate the closest external SaNTA entity
- Provide an IPSec-like security between each end-node/network and the closest external SaNTA entity

SaNTA security requirements:

Secure added value (2/2)

- Provide an IPSec-like security between SaNTA entities
 - With the widest end-to-end security policy for protecting transport payloads
 - With a security policy for protecting the negotiation of ITL connections between SaNTA entities
- Protection of radio-broadcasted traffic
 - Three possibilities:
 - Link security in low-level satellite communication protocols
 - Network layer security between the IWUs (e.g. IPSec)
 - Rely on SEs to properly address the security of traffic routed through satellite links
 - They can all be used together
 - But the latter alone is best for performance

SaNTA security requirements:

Correct exploitation

- Protection from abuses
 - Unauthorized access attempts
 - Denial-of-service attacks
- Basic authentication and authorization policy
 - SEs should only communicate with authorized peers
 - Authorized end-nodes and well-known IWUs
 - IWUs should only communicate with authorized peers
 - Well-known/authorized SEs and IWU peers
- Mechanisms for enforcing the policy in SEs and IWUs
 - Packet-filtering firewall for dropping unintended traffic
 - Strong authentication mechanism for authenticating IP peers
 - Example: IPSec authentication (preferably with AH)

SaNTA security:

Basic functionality

- The confidentiality is provided by SEs only
 - To provide confidentiality of data across internet networks and in traffic radio-broadcasted by the satellite
 - Only one encryption and decryption per payload
- Data encrypted by SEs
 - TCP payloads encapsulated in ITL
 - Alternatives: SSL, SSH, or IPSec
 - Information from original TCP headers used in the negotiation of ITL connections
 - Alternatives: SSL, SSH or IPSec
 - Original UDP datagrams encapsulated in ITL datagrams
 - Alternatives: IPSec
- All SaNTA entities use packet-filtering firewalls
 - To protect themselves from Internet hosts
 - To restrict incoming traffic to authorized end-nodes and peer SaNTA entities

SaNTA security architecture:

Topological deployment

■ SEs

- Should belong to the infrastructure of SaNTA clients
 - Within an isolated DMZ
 - On network gateways
- Should be completely or as much as possible managed by clients
 - End-networks administrators choose the right approach to route authorized traffic into SaNTA through a local SE
 - Should be protected from inside or outside attacks
 - To prevent usage abuses of satellite links imputable to SEs' owners

■ IWUs

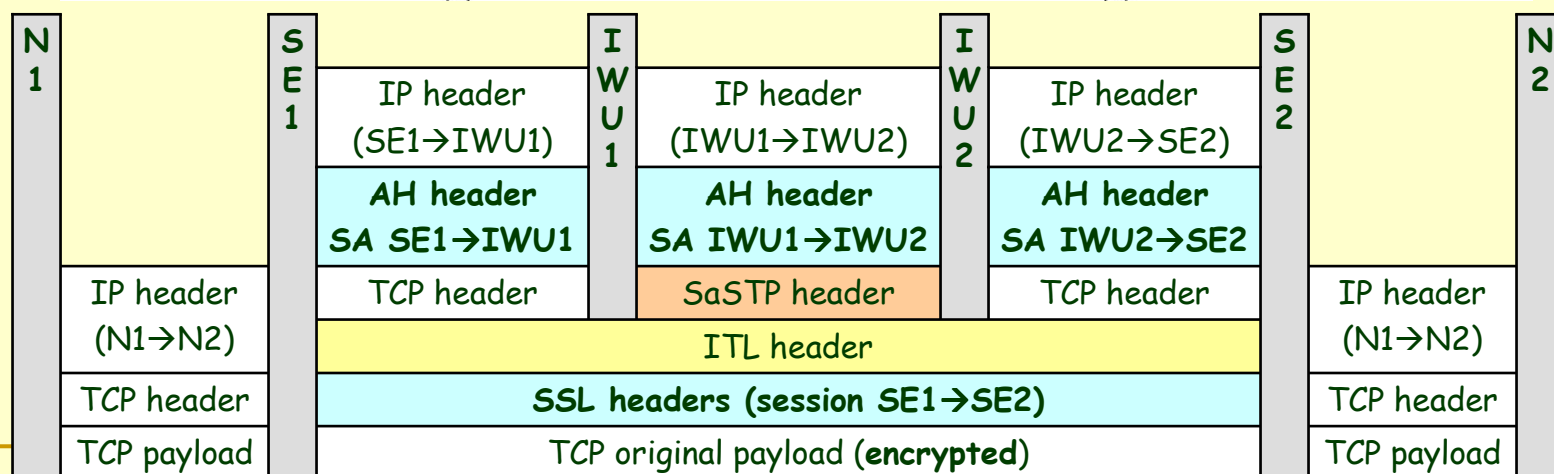
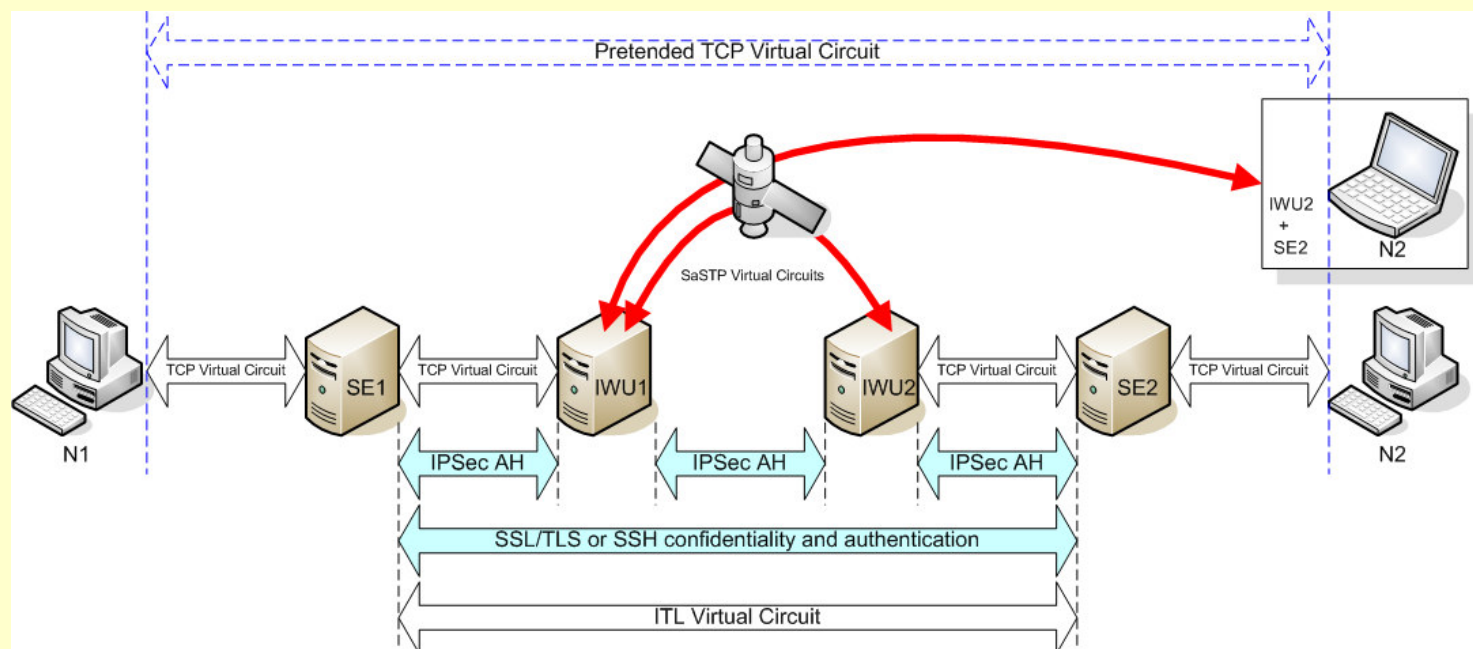
- Deployed and managed by satellite link providers
- Behave as Internet routers with SaNTA capabilities only for authorized SEs

SaNTA security architecture:

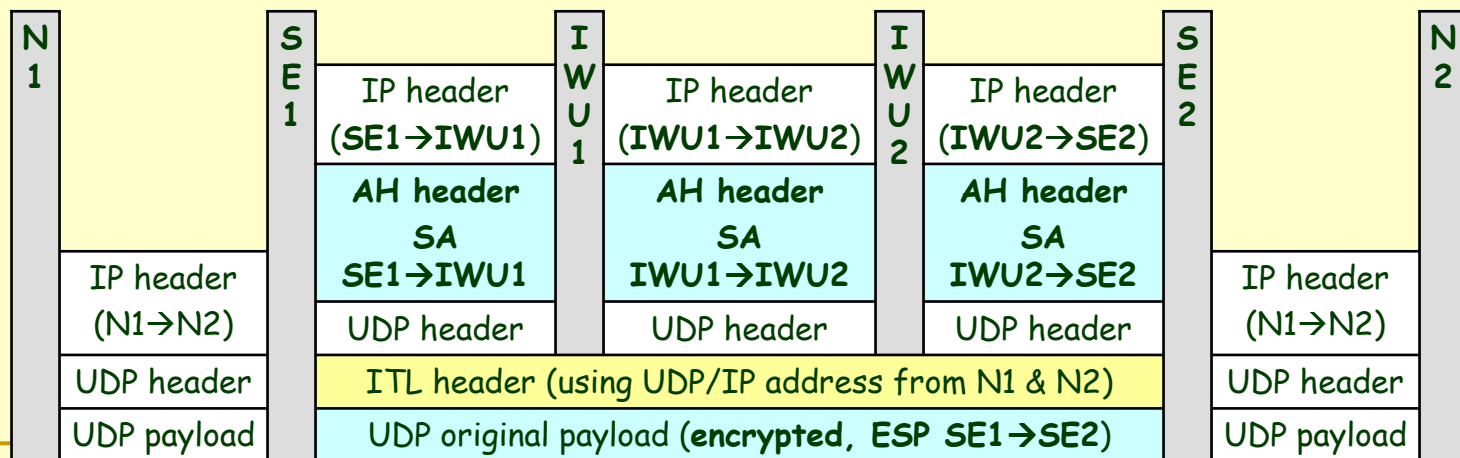
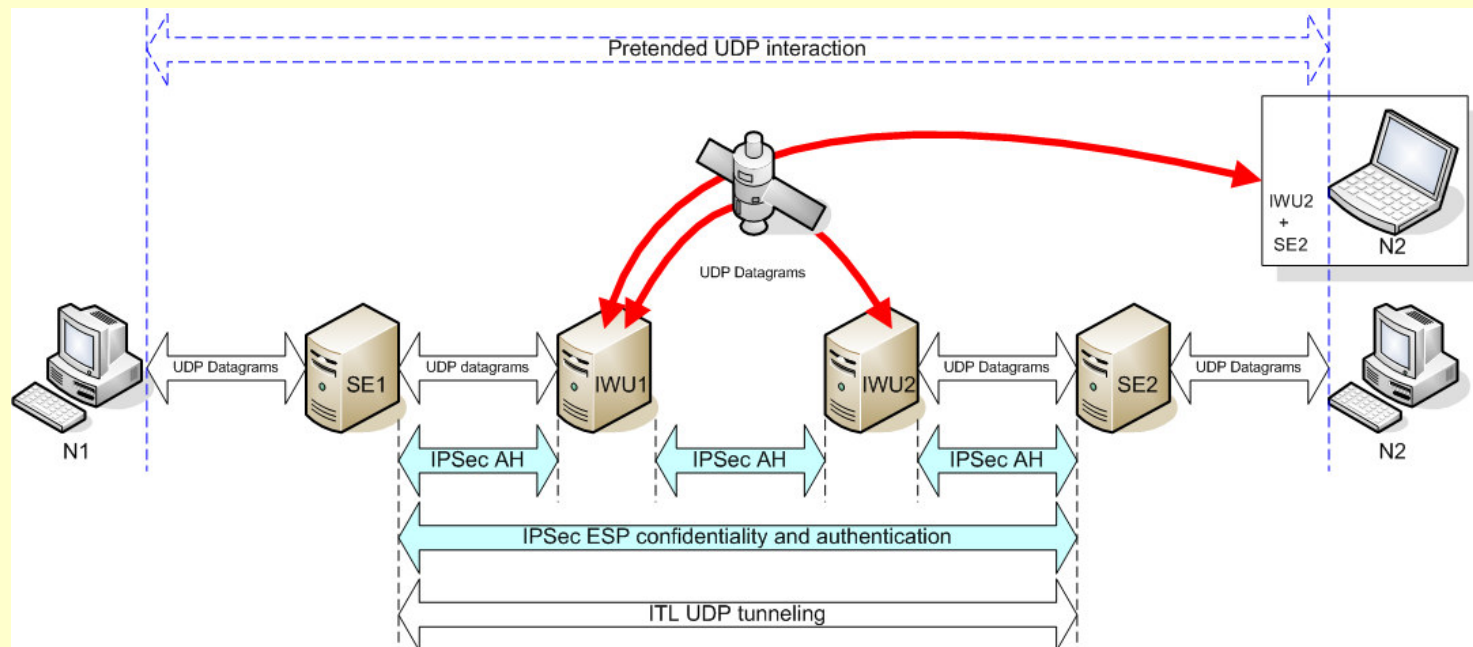
Authentication and access control

- End-hosts/networks \Leftrightarrow SEs
 - Managed solely by the administrators of end-networks
 - They all belong to the same security domain
 - Each SE should interact only with an authorized set of hosts within the end-network it belongs to
- Between SaNTA entities
 - With asymmetric cryptography and X.509 certificates
 - Issued by a self-certified CA managed by a satellite link owner and SaNTA provider
 - Facilitate long-term authentication and authorization of a consistent set of SaNTA entities within several secure communication protocols
 - Examples: IPSec and SSL
 - With secret, symmetric pre-shared keys
 - Can be used to authenticate IPSec peers (SEs and IWUs)
 - Cannot be used to authenticate SSL peers (SEs)
 - Badly chosen passwords can be subject to off-line dictionary attacks

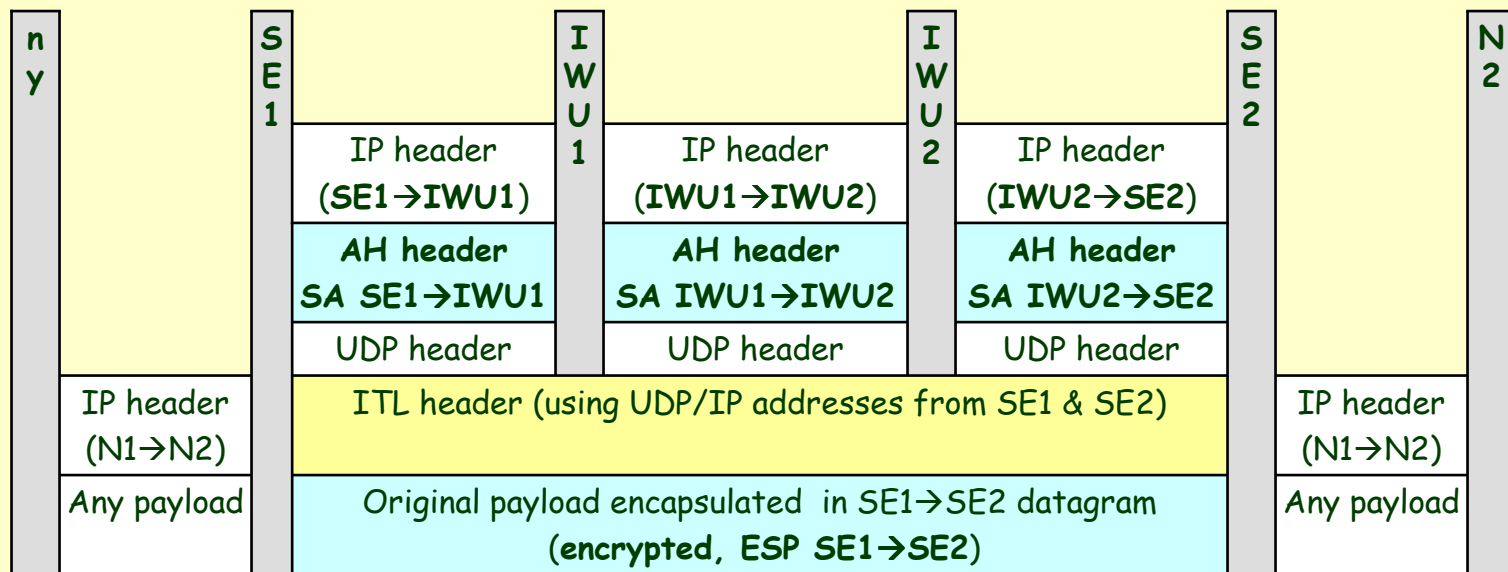
SaNTA security architecture: TCP data flows



SaNTA security architecture: UDP data flows



SaNTA security architecture: Opaque data flows



SaNTA Security:

Implementation details

- Implemented in Linux systems using standard features
 - OpenS/WAN IPSec modules & management tools
 - OpenSSL library
 - iptables features & libipq library
 - Standard Linux kernels and libraries
- Main implementation tasks
 - ITL upgrade
 - Add SaNTA tunneling for opaque data flows
 - Add SSL session management to ITL
 - Add IPSec SA negotiation and exploitation to SaNTA execution environment
- Implementation issues
 - Its hard to enforce two different IPSec SAs on a "mutating" IP datagram using only one machine

Conclusions

- The security architecture effectively protects remote interactions through a satellite link
 - SEs provide a proper VPN
 - Capable of accelerating TCP through a satellite link
 - End nodes/networks do not need to use other end-to-end security mechanisms
 - Allowing them to benefit from SaNTA acceleration
- SaNTA can handle all traffic protected with end-to-end mechanisms
- The SaNTA infrastructure is protected from abuses
 - Can only be used by authorized end nodes/networks
 - Authorized traffic is easy to identify and to validate

Future work

- Missing security protections
 - Protect ITL negotiations with SSL sessions
- Optimizations
 - Improve the efficiency in the negotiation of SSL sessions used by individual ITL connections
- QoS management
 - Differentiate the security applied to different TCP streams
 - For instance, end-to-end SSL sessions may not need to be further protected