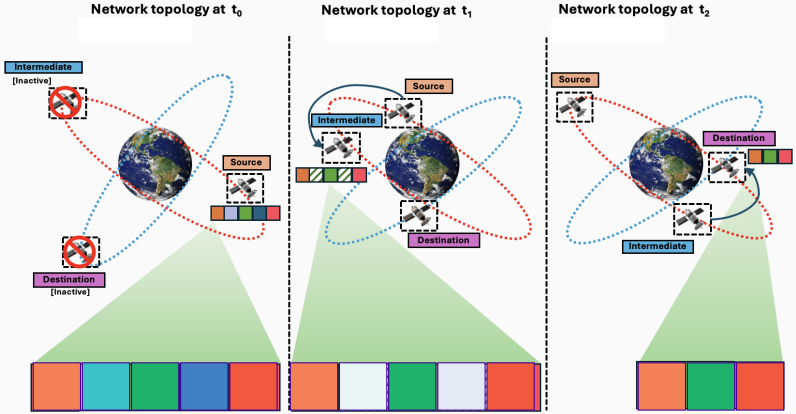
Detailed Security Considerations for Deepspace

Benjamin Dowling
benjamin.dowling@kcl.ac.uk

November 06, 2025

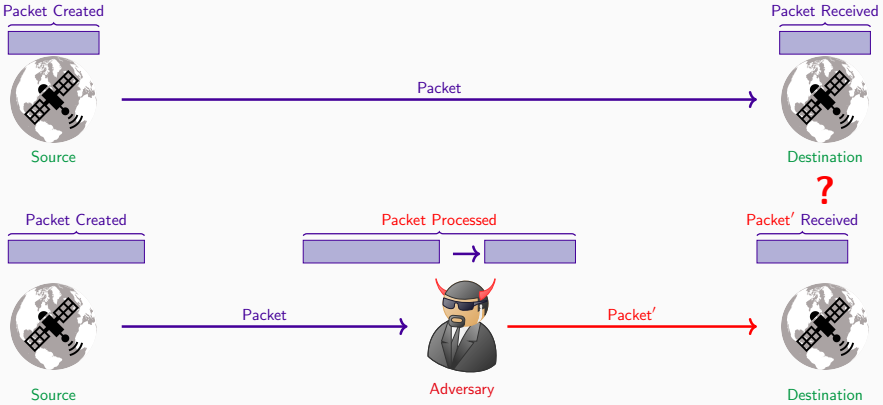
Functional Requirements

"Until full coverage..., the orbiters and other assets that are facing intermittent communications should provide store-and-forward capability." - **IP in Deep Space: Key Characteristics, Use Cases and Requirements**



Impacts on Security?

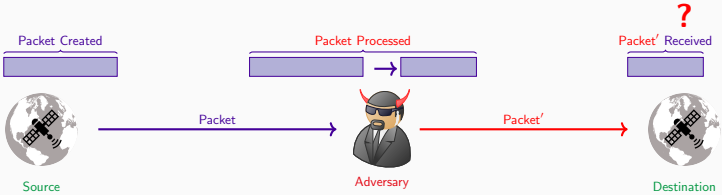
Need a mechanism to ensure confidentiality / authenticity.



But these properties can be quite fine-grained. Will address soon!

Realistic Threat Model?

"... over time it is possible that connectivity to the Internet may become available for some mission assets, and eventually interplanetary networks could become part of the Internet."

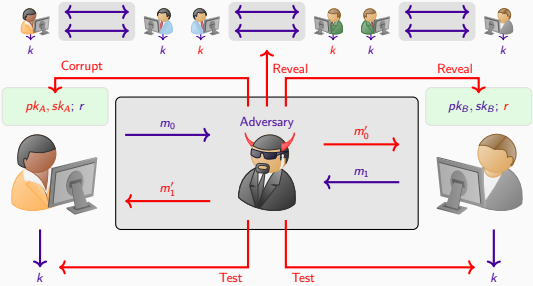


"Space exploration is more than ever carried by multiple stakeholders. A mixture of assets operated by government, commercial, and academic/research organizations from multiple nations are deployed." - **IP in Deep Space: Key Characteristics, Use Cases and Requirements**

Threat Modelling for Space

Security properties inform the design of the protocols we will use. What is an appropriate threat model? Typically, in the analysis of real-world cryptographic protocols (TLS, SSH, Signal, etc) we tend to consider security against an attacker that:

- is in complete control of the network,
  - Cryptographic modelling norm for network adversaries
- can **Reveal** derived message keys (k),
- can **Corrupt** long-term secrets (pk, sk),
- can **Compromise** short-term secrets (r)



Threat: Network adversary capable of compromising one or more of the endpoints. Security should be resilient against such an advesary.

Security Considerations in Current Draft

"Security protocols and mitigations are most often based on time, such as keys and certificate lifetimes, tokens lifetime, firewall opened ports time windows, etc. Keys and certificates have also a lifecycle where they should be renewed.

Given the delays to act, for example from Earth to a celestial body, notifications of security issues and mitigation actions will be delayed, which may increase the impact of security issues such as denial of service attacks.

Security validation, such as certificate validation, based on on-line immediate queries to validation databases, such as done with OCSP[RFC6960], or with a certificate chain will likely take too long if the validation database is far from the validation querier. Techniques such as preemptive caching of validation chain and certs should be investigated.

Certificate lifecycle should be well planned to take into account delays and intermittence, so that for example certificate renewals are done sufficiently in advance."

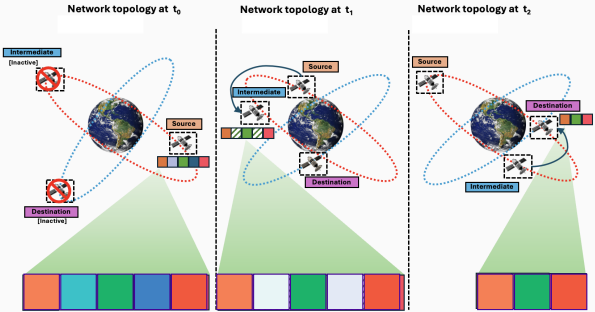
Mostly talking about long-term key management and not updating keys or considerations for doing so in space. What are the expected security properties of the channel?

Space Setting Challenges

"Compared to Internet on Earth, deep space communications and networking have multiple challenges, such as:

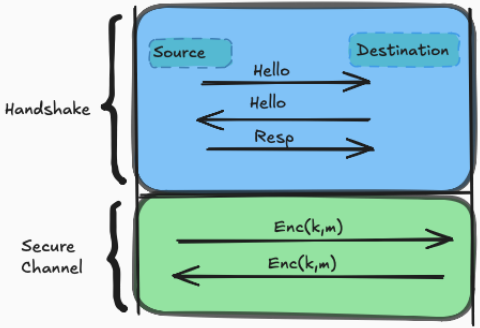
- *Significant and variable delays (e.g. ... multiple seconds, minutes, or hours).*
- *Frequent and long interruptions of communications, often with no alternate path."*

- IP in Deep Space: Key Characteristics, Use Cases and Requirements



Handshakes

To achieve confidentiality and authentication, we need to establish keys for secure communication. In secure communication these are typically called handshakes



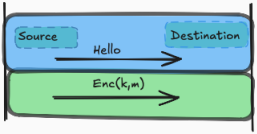
Any delay or dropping of packets prevents key establishment.

Thus, delays or dropping of handshake messages prevents secure payload communication.

Impact of Packet Loss on Handshakes

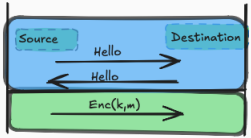
Different types of handshakes give us different security properties, but they also affect performance and functionality. With notably delays in space, there are operational impacts of security choices.

0-RTT



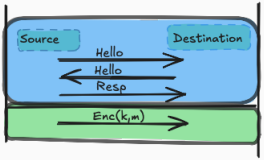
- asynchronous

0.5-RTT



- synchronous

1-RTT



- synchronous

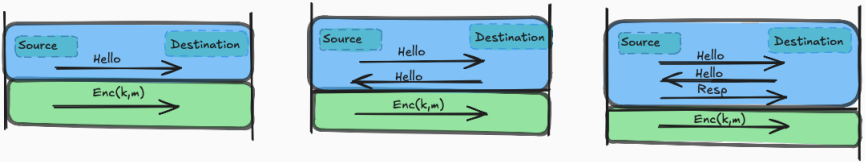
Let's assume you have some probability of packet loss P_{pl}. Logically, as the number of messages in the handshake increases, then so too does the probability of the handshake failing due to packet loss: $(1 - P_{pl}), (1 - P_{pl})^2, (1 - P_{pl})^3, \ldots$

Impact of Packet Loss on Handshakes... in Space! (ref `draft-tiptop-usecase` PR #43)

"Compared to Internet on Earth, deep space communications and networking have multiple challenges, such as:

- *Significant and variable delays (e.g. ... multiple seconds, minutes, or hours).*
- *Frequent and long interruptions of communications, often with no alternate path."*

- IP in Deep Space: Key Characteristics, Use Cases and Requirements



Packet loss is even more difficult to address in space setting, since the delays and interruptions hide whether packet loss occurred (and lead to incomplete secure channel establishment). Re-starting the handshake takes longer, and a new handshake might also be impacted by packet loss!

Async vs Sync Handshakes (ref `draft-tiptop-usecase` PR #43)

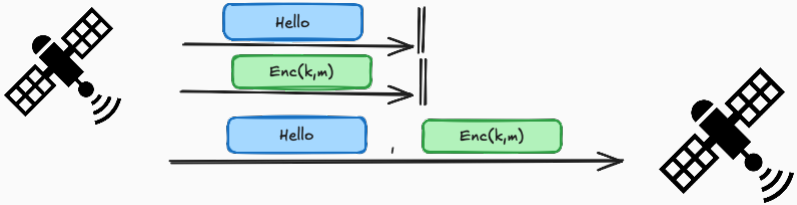
Synchronous handshakes require both entities to be responsive and have full handshake / act on updates with interactive and dependent contribution from both sides before a secure channel can be established.

- Packet loss can lead to handshake messages not being resolved. This can lead to multiple attempts to re-start a secure channel and/or an inability to send protected data.
- Packet loss can also lead to the last receiver in the synchronous handshake not receiving the final packet while the send assumes that the key update is ready to act on. This can lead to protected data being sent that cannot be processed on receipt.
- Long delays make this situation undesirable, which can drive an additional key confirmation step to avoid unprocessable data.

Notably, even minor packet loss in space can therefore result in significant delays and long interruptions from the security protocol design requirements - *even without additional incurred delays from the network!*

Async vs Sync Handshakes (ref `draft-tiptop-usecase` PR #43)

Asynchronous support allows for entities to be offline, thus massive delays in handshake messages can be resolved.



Asynchronous handshakes allow for key establishment (and key refresh, more on that later) without both parties being interactive.

This allows for resilience against packet loss during key establishment.

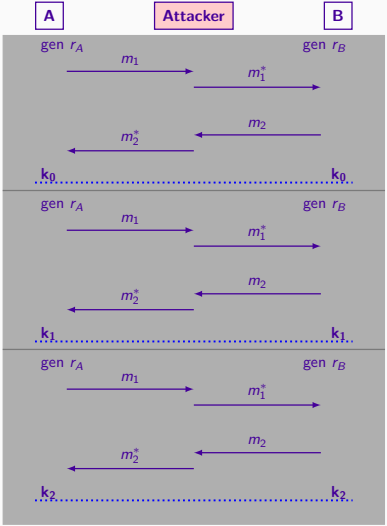
Fine-grained Security Considerations (ref `draft-tiptop-usecase` PR #45)

Secure Channel protocols today realise advanced security properties:

- **P**erfect **F**orward **S**ecrecy
- **P**ost **C**ompromise **S**ecurity

Consider an typical handshake (think the QUIC handshake protocol) executed in sequence between Alice and Bob

- Alice and Bob have long-term sk_A, sk_B
- Alice and Bob generate short-term values r_A, r_B during protocol execution
- Attacker has full network control



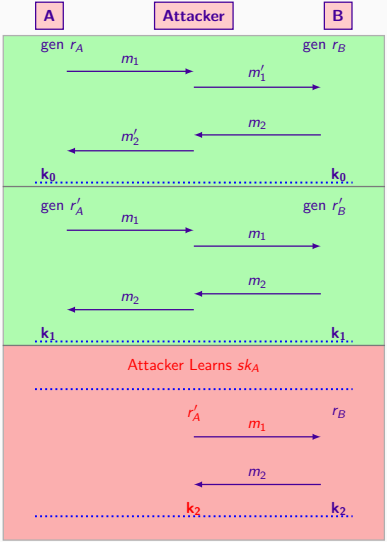
A generic key establishment protocol (e.g., QUIC)

Background: Perfect Forward Secrecy (ref `draft-tiptop-usecase` PR #45)

PFS considers the following scenario:

- Attacker has full control of the network
- Alice and Bob execute the handshake together
- Attacker **Corrupts** keys
- Attacker uses this to impersonate Alice

Obviously, protocol executions with "Alice" after this point are not secure - but what about before Attacker learns the keys? If confidentiality holds, the protocol has Perfect Forward Secrecy

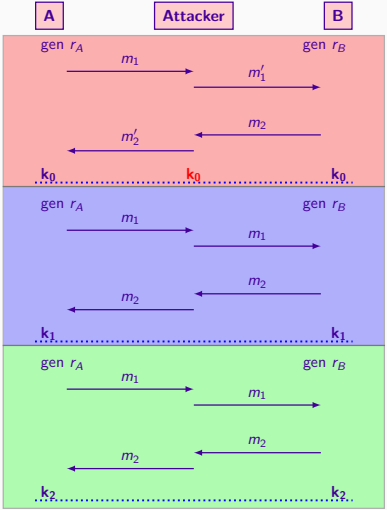


Background: Post Compromise Security (ref `draft-tiptop-usecase` PR #45)

PCS considers the following scenario:

- Attacker has full control of the network
- Attacker **Corrupts** and **Compromises** the full state of Alice (or Bob)
- Attacker becomes passive
- Alice and Bob execute channel establishment without modification or adversary interference

If Alice and Bob can recover security as a result of this passivity, preventing the attacker from accessing future data, then the protocol has post-compromise security

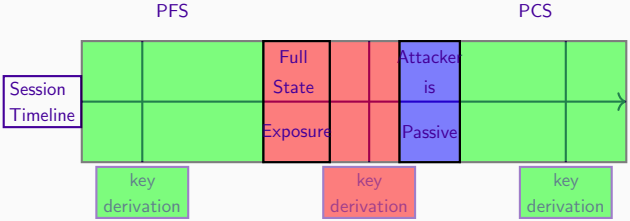


Security Considerations and TIPTOP (ref `draft-tiptop-usecase` PR #45)

> *"For endpoints, applications may keep a state context in permanent memory, either at the application layer or at the transport layer or both."* - **IP in Deep Space: Key Characteristics, Use Cases and Requirements**

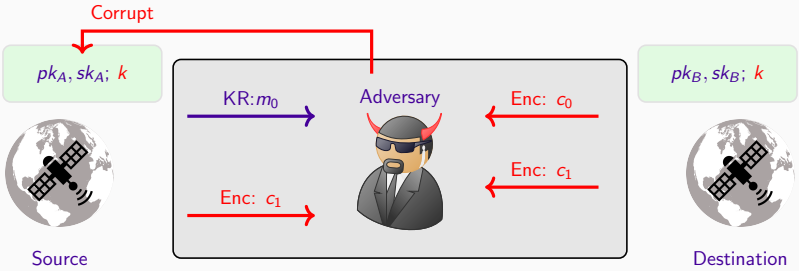
TIPTOP is designing for environments using secure channels, where transport session information may be stored permanently in state (lengthening the window of impact of device-compromise adversaries). Security considerations should reflect this.

PFS has become an expected security expectation for all use cases, and PCS is quickly also spreading towards normative.



Challenges Impact on Security

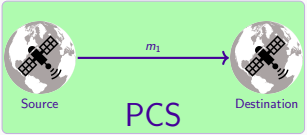
Any security action (i.e., handshake) that requires communication to protect against compromise may be subverted via packet loss or packet delay.



If a handshake packet is dropped, any adversary that compromised secrets can continue to read all communication. Let's look at the impact on PFS/PCS.

How To Achieve PCS/PFS

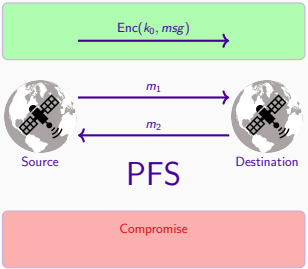
Any key establishment protocol that achieves PFS/PCS requires key refresh, i.e. typically established by **communication** between both parties via handshakes.

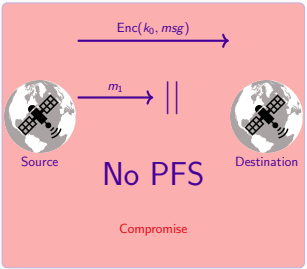


Sometimes this communication is unavoidable (need to establish new secrets via KEX), sometimes it can be avoided. Thus, best to achieve such guarantees in a round-optimal fashion. Why?

Impact of Delay/Dropping on Security

Any delay in a handshake extends the window/impact of compromise.



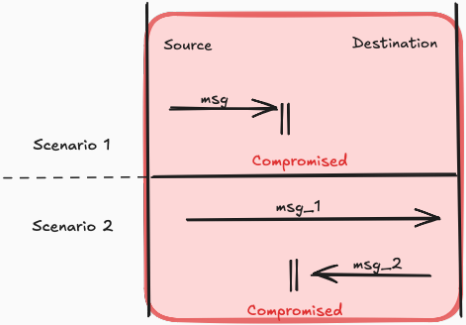


More communication needed to establish goals gives more changes for delays to key refresh, and thus extends the impact of compromise.

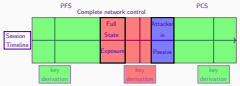
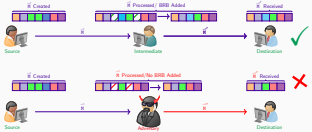
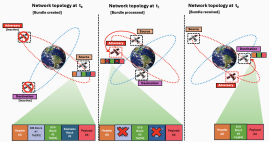
Communication Overhead and Impact on Security

Any security properties that require that the handshake first execute a full round of communication are more strongly impacted.

Delays on either Hello/Response message **prevent key refresh**.



Requirements Document

- Highlight the impact of the delays and interruptions on security design choices and need for support that is different in space vs terrestrial
- Include perfect forward secrecy and post-compromise security as core security requirements
- Ensure that designs can support enclave and group communication topologies in an efficient manner under the space constraints without compromising security
- Ensure requirements to support quantum resistance as is the norm across the IETF so that design choices are future proof
- Could expand further on security properties required, specifically, the interaction of the key characteristics (interruptions, delays, packet loss) with the security (key establishment, key update, detailed security properties)

Others?