SplitPad: Securing Communication with Active Networking

- ☞ A particular application scenario.
- ☞ 100% bullet-proof communication (perfect privacy).

One-time Pads

- ❑ The only know encryption that is proven to be unbreakable.
- ❑ Encoding algorithm for N bits: $\overline{m_i}; i = 1 \ldots N; m_i \in \{0, 1\}$.
  - – Be $\overline{r_i}$ a sequence of (equally distributed) random bits.
  - – Calculate the ciphertext $\overline{c_i}$: $c_i = r_i \otimes m_i$ (bitwise xor).
  - – Keep $\overline{r_i}$ secret.
- ❑ Decoding: Calculate $\overline{d_i}$:$d_i = r_i \otimes c_i = r_i \otimes r_i \otimes m_i = m_i$.
- ❑ Destroy $\overline{r_i}$.
  - ☞ One-time pad: $\overline{r_i}$ can be used only once for decoding.
  - ☞ Application: secret sharing, in weakened form: OFB mode.

Security of the One-time Pad

- ❑ It is impossible for the cryptanalyst having only ciphertext $\overline{c_i}$ to calculate the message $\overline{m_i}$.
  - – For each cipherbit c_i, the probability that the original was a 0 (resp.1) is exactly 0.5.
  - – For a given c_i, every possible $\overline{m'_i}$ has exactly the same probability 2^{-N}.
- ❑ “The key $\overline{r_i}$ is as long as the message itself, and chosen carefully.”

The Problem with the One-time Pad

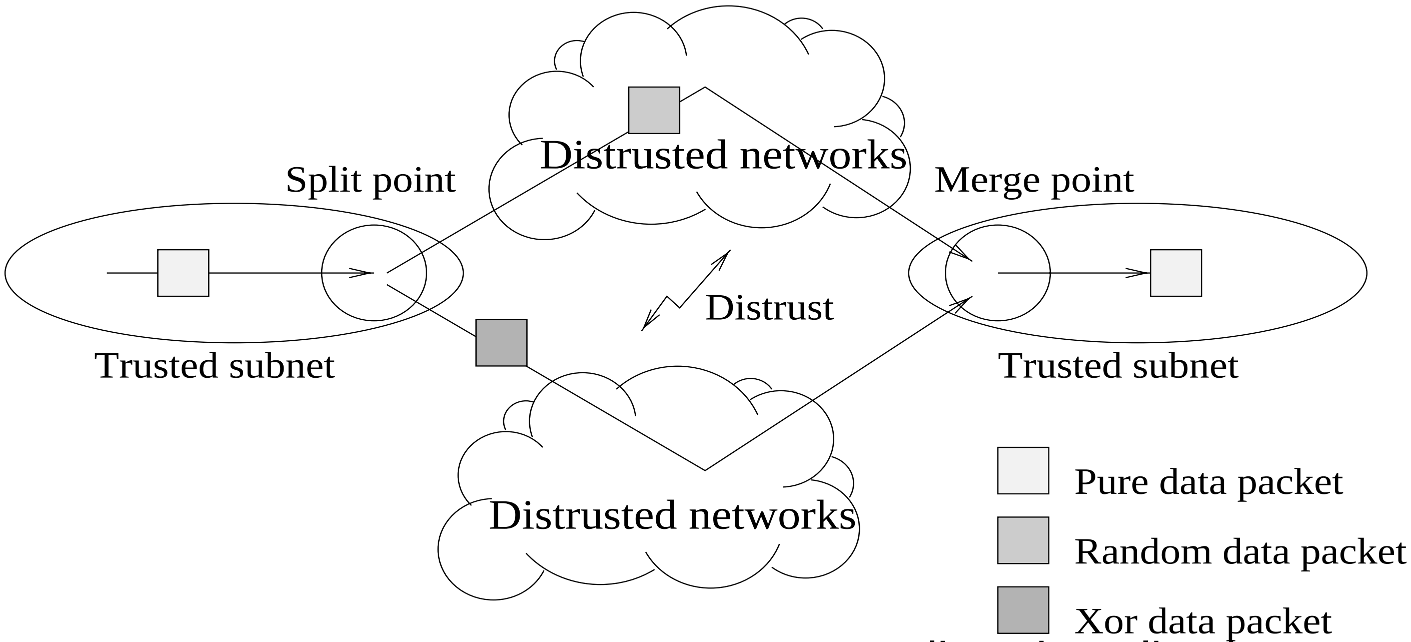
- Where to get that much ‘good’ random?
  - Mechanical random e.g. lottery machines, dices.
  - Physical random e.g. radioactive decay.
  - Human behaviour e.g. keyboard interrupt times.
  - Multiprocessing & *networking devices*:

```
(ps -el & netstat -na & netstat -s & ls -lLRt /dev & w) | md5
```

- How to bring $\overline{r}_i$ to the receiver?
  - Use an independent communication infrastructure:
  - ☞ Postal service, Telephone, messenger.
  - *Use independent network paths*:
  - ☞ Different ISP, different physical links, paths through different countries.

Active Networking: enabling SplitPad

- The capsules implement SplitPad.
- Dynamic setup of independent paths.
- Dynamic setup of the ‘split-point’ (resp. merge-point).
- Dynamic deployment of necessary transport layer protocol.
- Dynamic deployment of random generating code.
  - Make use of e.g. packet latency and the state of the network node.



SplitPad application scenario.

Particular Problems

❑ Setup the paths.

- ☞ Pathfinder capsules.

❑ Delays & loss of split capsules.

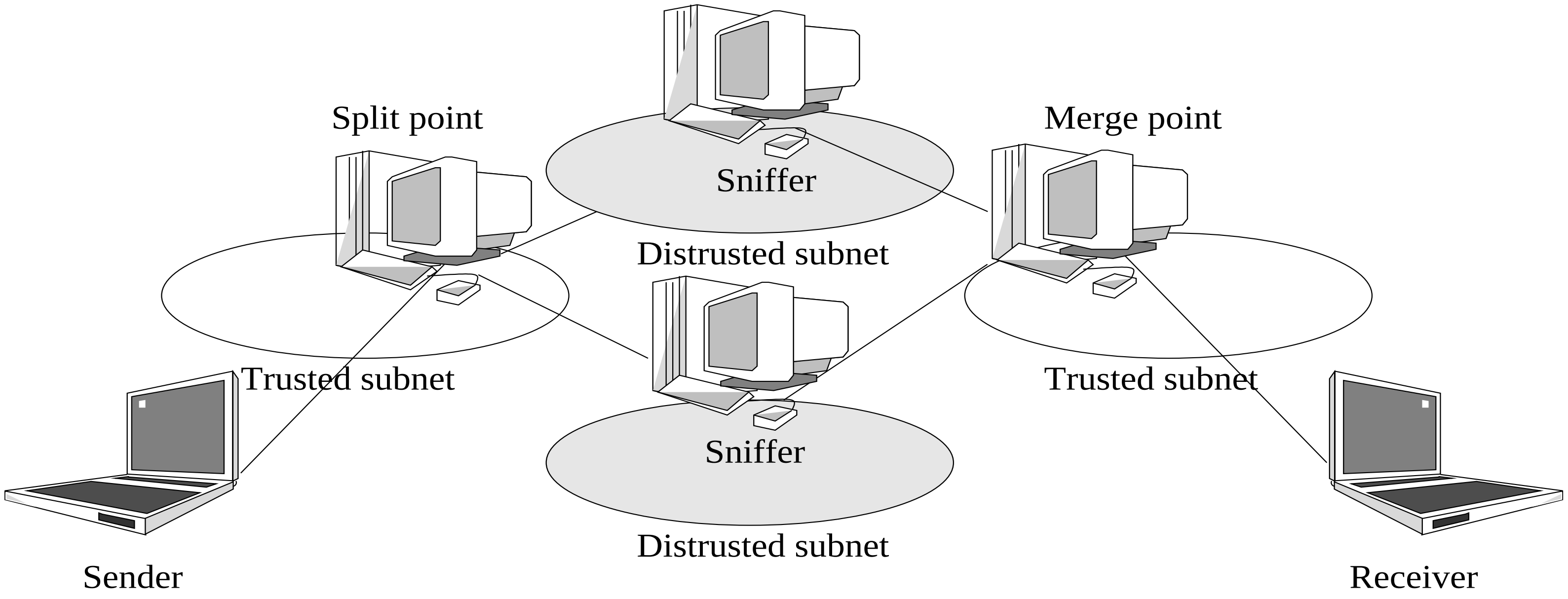
- ☞ Split-capsule has code to wait for its twin at the merge node.

❑ Generation of ‘sufficient’ random.

- – The delay variation provides only few random bits (limited clock resolution).
- ☞ Bootstrapping with empty capsules.
- ☞ Use a secure random number generator.
- ☞ The generator should only work with a large seed (>128 bits).

Conclusions

- ❑ Active networking allows the dynamic deployment of the SplitPad scheme.
  - Application of the well-known one-time pad.
  - High level of data communication privacy for specific application areas.
  - Computational light weighted especially on the receiver side.
- ❑ Implementation with the Active Node Transfer System (ANTS) of the MIT.
- ❑ Demo setting:


Split point
Sniffer
Merge point
Distrusted subnet
Trusted subnet
Trusted subnet
Sniffer
Distrusted subnet
Sender
Receiver