Yue-bin Luo, Bao-sheng Wang, Xiao-feng Wang, Bo-feng Zhang, 2017. A keyed-hashing based self-synchronization mechanism for port address hopping communication. *Frontiers of Information Technology & Electronic Engineering*, **18**(5):719-728. http://dx.doi.org/10.1631/FITEE.1601548

A keyed-hashing based self-synchronization mechanism for port address hopping communication

Key words: Synchronization; Port address hopping; Moving target defense; Network security

Corresponding author: Yue-bin Luo
E-mail: luoyuebin@nudt.edu.cn

ORCID: http://orcid.org/0000-0002-8194-5262

Motivation

- Port address hopping (PAH) communication is a powerful network moving target defense (MTD) mechanism inspired by frequency hopping in wireless communications.
- One of the critical and difficult issues involved in the design of PAH systems is synchronizing the communication parties.
- Existing schemes usually provide hops for each session lasting only a few seconds/minutes, making them easily influenced by network events such as transmission delays, traffic jams, packet dropouts, reordering, and retransmission.
- A new synchronization mechanism adapting to complex network situations without using third-party servers is needed.

Main idea

- We introduce the HMAC message authentication mechanism and design a self-synchronization mechanism for PAH, called keyed-hashing based self-synchronization (KHSS).
- In KHSS, a message is hashed to a digest which is then used for port and address encoding/decoding in PAH.
- This enables us to conceal, from layers 2 and 3's perspectives, the identity of the actual server machines and services. Such concealment is sufficient to defeat a large pool of network-level traffic analysis tools.

Method

1. KHSS requires a cryptographic hash function denoted by *H* (e.g., MD5, SHA-1), in combination with a secret key *K* which can be achieved using a secret handshake scheme (Gu and Xue, 2011).
2. Every time a PAH converter receives a message, the message is used as the input for HMAC, and HMAC generates an L-bit hash sum (L=128 for MD5, L=160 for SHA-1) as its output.
3. Then, the hash sum is used to perform exclusive OR operations with the communication identities that are extracted from the message and determine the communication identities to which they should be changed.

Method

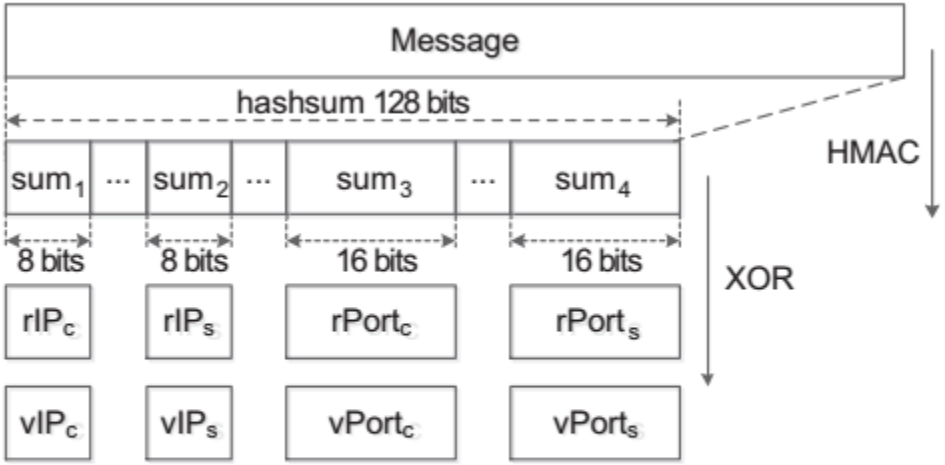

Fig. 2 General structure of the keyed-hashing based self-synchronization scheme (HMAC: hash based MAC)

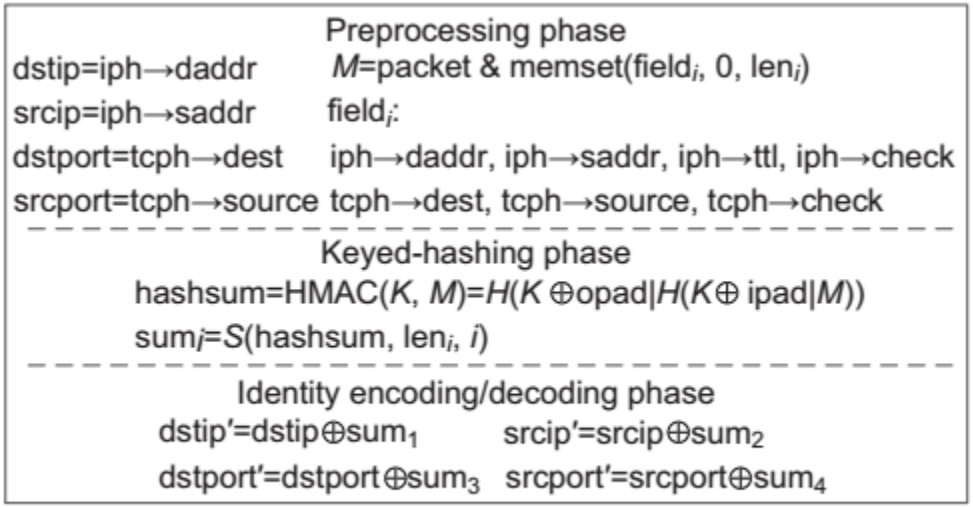

Fig. 3 Keyed-hashing based self-synchronization operations in detail

Major results

- KHSS-based PAH system provides the maximum hopping frequency

Table 1 Comparison of the hopping times

Scheme	Hopping period (100 Mb/s Ethernet)
RPH	UDP: 0.5 s; TCP: 12–19 s
PRC	At least one round-trip time (168.9 ms)
DL/PC-based PH	1–4 packets time (6.72–26.88 μs)
NASR	15 min
RPAH	5 s
KHSS-based PAH	One packet time (6.72 μs)

PH: port hopping; RPH: random PH; PRC: port rationing channel; DL: data length; PC: packet count; NASR: network address space randomization; PAH: port and address hopping; RPAH: random PAH; KHSS: keyed-hashing based self-synchronization

Major results

- KHSS-based PAH mechanism can provide a more effective and active defense method for the host and service hiding compared to the existing schemes

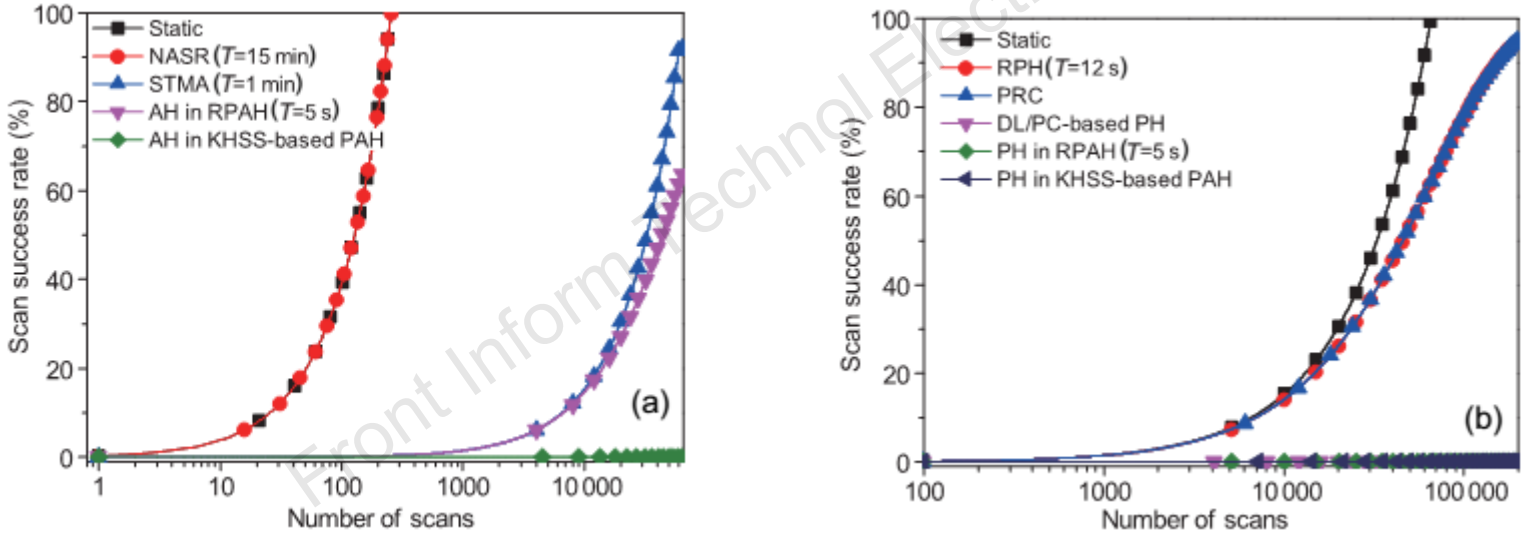

Fig. 4 Scan success rate of different address hopping (AH) schemes (a) and different port hopping (PH) schemes (b) vs. the number of scans (NASR: network address space randomization; PAH: port and AH; RPAH: random PAH; KHSS: keyed-hashing based self-synchronization; RPH: random PH; PRC: port rationing channel; DL: data length; PC: packet count)

Major results

- KHSS-based PAH mechanism outperforms existing hopping schemes and provides sufficient communication and synchronization performance

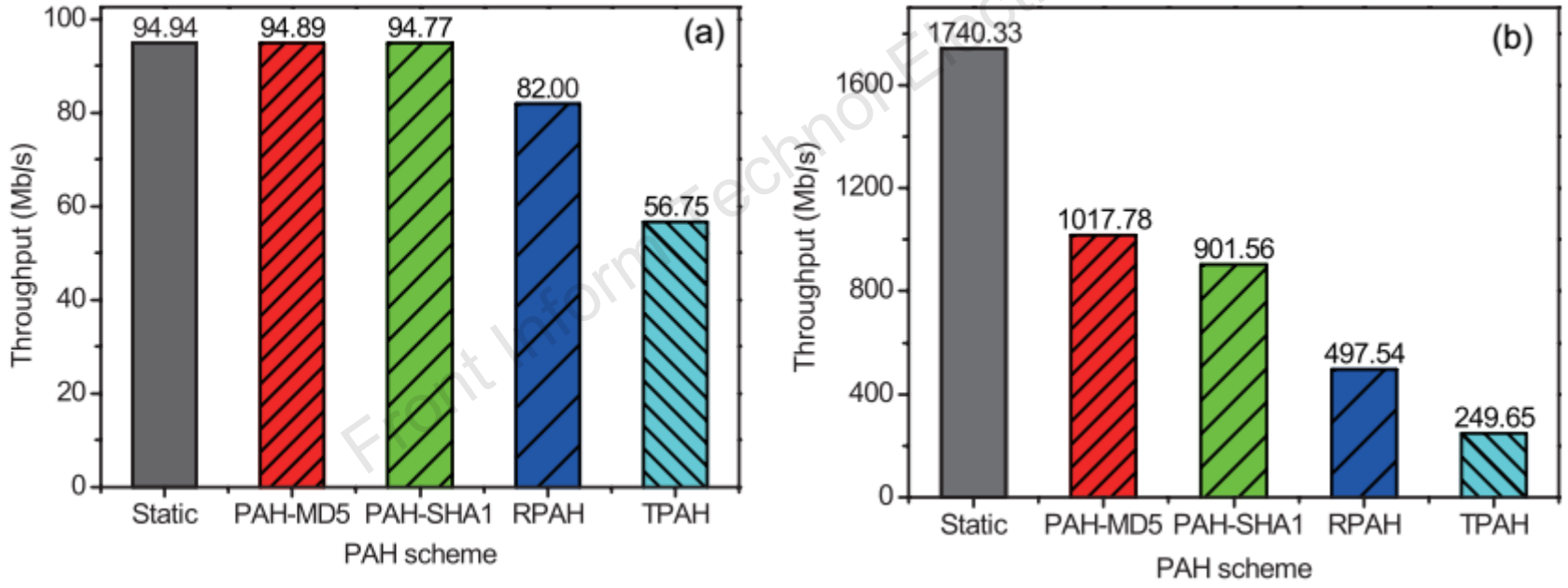

Fig. 5 Results of the throughput tests on our campus network (a) and on a virtual network (b) (PAH: port address hopping; RPAH: random PAH; TPAH: TAP-based PAH)

Conclusions

- KHSS enables one-packet-one-PAH and invisible message authentication schemes without synchronization information and message authentication code transmissions.
- KHSS presents significant advantages over existing synchronization schemes in terms of both security and hopping efficiency.
- The overhead when introducing the KHSS mechanism in a traditional network is quite low (below 0.2%) and KHSS is good enough to be deployed in most of application scenarios in a high bandwidth network.