

# Data Path Credentials for High-Performance Capabilities-Based Networks

Tilman Wolf

Department of Electrical and Computer Engineering  
University of Massachusetts, Amherst, MA 01003  
wolf@ecs.umass.edu

## ABSTRACT

Capabilities-based networks present a fundamental shift in the security design of network architectures. Instead of permitting the transmission of packets from any source to any destination, routers deny forwarding by default. For a successful transmission, packets need to positively identify themselves and their permissions to the router. The analysis of the data path credentials data structure that we propose shows that as few as 128 bits are sufficient to reduce the probability of unauthorized traffic reaching its destination to a fraction of a percent.

## General Terms

Design, Security

## Categories and Subject Descriptors

C.2.1 [Computer-Communication Networks]: General—*Security and protection*

## 1. NETWORKS WITH CAPABILITIES

Recent proposals for capabilities-based networks have suggested a fundamental shift in the design philosophy of the Internet by moving from an “on-by-default” to an “off-by-default” assumption. The initial idea has been introduced by Anderson et al. [1] in the context of DoS attacks and further explored by Ballani et al. [2] for other attacks. In such a network, a connection needs to be explicitly authorized to reach an end-system rather than being allowed to connect to an end-system by default. Authorization is based on capabilities, which are tokens that represent authority for a particular operation. During the connection setup and data transfer, a connection’s capabilities are validated along the connection path.

We present a capabilities-based network architecture with a novel design of capabilities, which we call “data path credentials.” In particular, these credentials can be validated easily in the data path of routers and thus allow high-

performance implementations. Therefore, we can check capabilities on every hop along the path and provide effective defenses against a range of attacks. In our prior work, we have introduced the general architecture of this network design [3] as well as initial ideas on how to design data path credentials [4]. A major challenge for a high-performance implementation of such a network is an efficient design of the credentials that are carried in the packet and the verification procedure on the router. We present a data path credentials data structure that is based on Bloom filters. The credentials are fixed length (independent of the number of routers that are traversed by the packet) and can be verified by routers with a few simple operations. Our analysis shows that credentials as small as 128 bits can effectively reduce the probability of unauthorized traffic reaching its destination to a fraction of a percent.

## 2. DATA PATH CREDENTIALS DESIGN

The data path credentials data structure that is used to carry packet permissions is based on Bloom filters. A Bloom filter is a bit array that can store  $m$  bits. Using  $k$  different hash functions  $h_1(x) \dots h_k(x)$ , an element  $x$  is mapped to  $k$  bit position in the array. When adding element  $x$ , the bits corresponding to the hash function values for element  $x$  are set to 1. When performing a check for membership of an element, the hash functions for the element are computed and it is checked if the according bits in the array are set. Only if all of these bits are set to 1, the element is reported to be a member of the set.

To use the Bloom filter data structure as data path credentials for packets that traverse the network, we store credentials from each router along the path. The source node of a connection negotiates permission to transmit data across a router during connection setup. When router  $j$  ( $1 \leq j \leq n$ ) permits transmission, it provides the source with its router credentials  $r_j$ . Router credentials are the set of indices  $r_j[i]$  ( $1 \leq i \leq k$ ) of bits that are set in the Bloom filter array. The credentials from all routers along the path are then superimposed (i.e., logical OR operation) in the Bloom filter data structure. This creates aggregate credentials  $c$  (consisting of a single bit array of size  $m$ ) that are sent with each data packet (see Figure 1). When receiving a packet with aggregate credentials  $c$ , router  $j$  can then check the value of all bits that were provided in router credentials  $r_j$ . If the aggregate credentials are valid, then  $\prod_i c[r_j[i]] = 1$ , where the product is the equivalent of a logical AND operation. If the aggregate credentials do not contain the router credentials of a particular router, it is likely that one of the bits in

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ANCS’08, November 6–7, 2008, San Jose, CA, USA.

Copyright 2008 ACM 978-1-60558-346-4/08/0011 ...\$5.00.

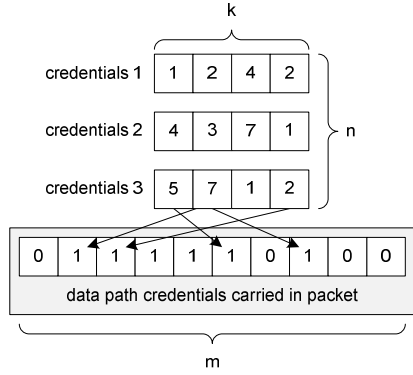


Figure 1: Credentials Data Structure.

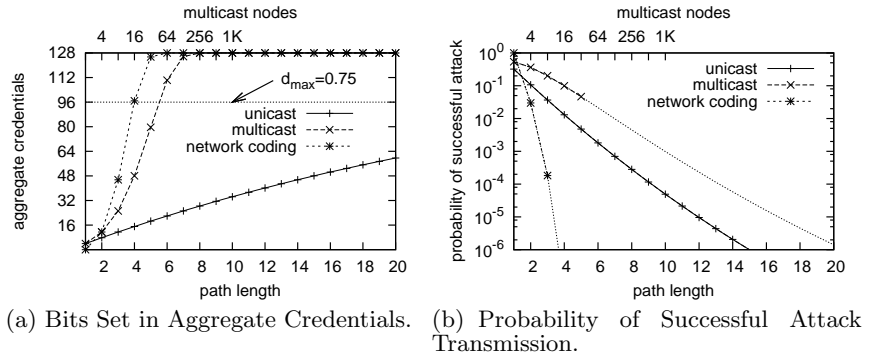


Figure 2: Evaluation of Data Path Credentials (size  $m=128$ , number of hash functions  $k=4$ ).

credentials  $c$  does not contain a 1 at one of the router credentials' bit positions. Thus the validation of the aggregate credentials fails.

The security of the network architecture depends on the security of the credentials. We can achieve this by using cryptographically strong hash functions (e.g., MD5 or SHA-1) where router  $j$  uses  $k$  secret keys  $s_j[i], 1 \leq i \leq k$ . The cryptographic hash function  $h_i(s_j[i], f)$  uses router  $j$ 's key for bit index  $k$  to determine which bits to set in the aggregate credentials. It is important that this function also uses flow identifier  $f$  (e.g., based on a 5-tuple hash) as an input to avoid attacks where credentials from an authenticated connection are used by a different connection. In order to make data path credentials immune to a simple attack, where all bits in the data path credentials are simply set to 1, we introduce a "density" metric  $d(c)$  that reflects the number of 1's in credentials  $c$  as a fraction of the total size:  $d(c) = \frac{1}{m} \sum_i c[i]$ . To consider credentials valid, we require that the density is equal or below a certain threshold:  $d(c) \leq d_{max}$ .

### 3. SECURITY ANALYSIS

Since a Bloom filter can yield false positives, it is possible that traffic with forged credentials may pass through the network. In a *unicast* scenario, attack traffic needs to traverse  $n$  hops from source to destination and pass credential checks on each hop. When validating credentials, a router checks if all  $k$  bits of its credentials are set. To reach the destination, all bits set (in valid credentials) are checked at least once. Thus, an attacker has to be able to create forged credentials with at least those bits set. The number of bits set,  $b(m, k, n)$ , in a Bloom filter of size  $m$  with  $k$  hash functions and  $n$  stored items is  $b(m, k, n) = m \cdot \left(1 - \left(1 - \frac{1}{m}\right)^{kn}\right)$ . A false positive transmission (i.e., a successful attack) occurs when among the bits set by the attacker (limited by  $d_{max}$ ), there is the correct set of  $b(m, k, n)$  bits that is checked by the set of routers along the path. The probability for this event is  $f_{unicast}(m, k, h) = (d_{max})^{b(m, k, h)}$ .

Similarly, the false positive transmission probability for *multicast/multipath* is  $f_{multicast}(m, k, h) = 1 - \left(1 - f_{unicast}(m, k, h)\right)^{2^h}$  (assuming a binary multicast tree of height  $h$ ). For *network coding*, where multiple packets are necessary to reconstruct the original transmissions, the probability is  $f_{network\ coding}(m, k, h) = \left(f_{multicast}(m, k, h - 1) \cdot f_{unicast}(m, k, 1)\right)^{2^{(h-1)}}$ .

Figure 2(a) shows the number of bits set (as determined by  $b(m, k, n)$ ) for credentials of size 128 bits and four hash functions. As the path length increases, the number of bits set by credentials also increases. For multicast and network coding, the increase is much steeper than for unicast since many more credentials are aggregated due to the larger number of paths and destinations. When the maximum density (in this example  $d_{max} = 0.75$ ) is exceeded, credentials are rejected by all routers. The false positive rates that correspond to this example are shown in Figure 2(b). Data points are only shown for those cases where the maximum density is not exceeded (the dotted lines continue beyond this limit to illustrate the overall trends). The decreasing trend with an increasing number of hops is due to repeated credential checks. The results show that adding credentials with as few as 128 bits to packets can reduce the probability that attack traffic can reach its destination to a fraction of a percent.

### Acknowledgements

This material is based upon work under subcontract #069153 issued by BAE Systems National Security Solutions, Inc. and supported by the Defense Advanced Research Projects Agency (DARPA) and the Space and Naval Warfare System Center (SPAWARSYSCEN), San Diego under Contract No. N66001-08-C-2013.

### 4. REFERENCES

- [1] ANDERSON, T., ROSCOE, T., AND WETHERALL, D. Preventing Internet denial-of-service with capabilities. *SIGCOMM Computer Communication Review* 34, 1 (Jan. 2004), 39–44.
- [2] BALLANI, H., CHAWATHE, Y., RATNASAMY, S., ROSCOE, T., AND SHENKER, S. Off by default! In *Proc. of Fourth Workshop on Hot Topics in Networks (HotNets-IV)* (College Park, MD, Nov. 2005).
- [3] WOLF, T. A credential-based data path architecture for assurable global networking. In *Proc. of the 2007 IEEE Conference on Military Communications (MILCOM)* (Orlando, FL, Oct. 2007).
- [4] WOLF, T. Design of a network architecture with inherent data path security. In *Proc. of ACM/IEEE Symposium on Architectures for Networking and Communication Systems (ANCS)* (Orlando, FL, Dec. 2007), pp. 39–40.