Network Working Group   C. Vogt
Internet-Draft   Universitaet Karlsruhe (TH)
Expires: August 18, 2006   J. Arkko
                                          Ericsson Research NomadicLab
                                                  February 14, 2006

  Credit-Based Authorization for Concurrent Reachability Verification
              draft-vogt-mobopts-simple-cba-00.txt

Status of this Memo

Copyright Notice

Abstract

   Mobility and multi-homing protocols enable multi-addressed nodes to
   redirect ongoing communication sessions from one IP address to
   another. Most of these protocols verify a multi-addressed node's
   reachability at a claimed new IP address in order to prevent
   redirection-based flooding attacks. In view of reduced protocol
   latencies, such verification is preferably performed concurrently,
   i.e., while packets are already being sent to the new IP address.

   This document defines Credit-Based Authorization, a technique that
   facilitates concurrent reachability verification without compromise
   of security.

Table of Contents

1. **Introduction**

   Mobility and multi-homing protocols enable multi-addressed nodes to
   redirect ongoing communication sessions from one IP address to
   another, be it for the purpose of mobility support, recovery from a
   failure upstream in the network, network renumbering, or a DHCP lease
   expiry. Many of these protocols operate end to end, and it is a
   multi-addressed node's responsibility to inform its correspondent
   node(s) when it changes IP connectivity.

   An undesired implication of end-to-end packet redirection is that,
   when a correspondent node learns that a multi-addressed peer has a
   new IP address, it does not necessarily know whether the peer is
   actually reachable at that IP address. In fact, a malicious peer may
   intentionally give a false IP address in order to cause a packet
   flood on the victim located there [6]. Likewise, viral software may
   have compromised the peer, programming it to redirect packets to a
   specific victim. Such redirection-based flooding is particularly
   serious due to its potential for high amplification: It is generally
   sufficient for the attacker to spoof small acknowledgments so that
   the correspondent node's instance of the transport protocol continues
   to send larger data packets to the victim. Similar means for
   amplification are today possible only through distributed denial-of-
   service attacks.

   Most mobility and multi-homing protocols mitigate the threat of
   redirection-based flooding by checking a multi-addressed node's
   reachability at a new IP address before data packets are sent to that
   IP address. For this, the correspondent node sends an unguessable
   number to the new IP address and waits for this number to be echoed
   by the multi-addressed peer. In its basic form, reachability
   verification requires the correspondent node to refrain from sending
   packets to the new IP address until the multi-addressed node is known
   to be present at that address. This causes undesired delays,
   however, which have been shown [7] to compromise the quality of real-
   time applications such as VoIP, video conferencing, and multi-player
   games, and to cause adverse reactions of TCP's retransmission and
   congestion-avoidance mechanisms. Concurrent reachability
   verification has been proposed as an optimization, but additional
   protection is necessary during the phase when packets are sent to a
   new IP address while it is yet unverified.

   This document specifies a mechanism, Credit-Based Authorization, that
   facilitates secure and concurrent reachability verification. The
   mechanism is fully transparent to the multi-addressed node. In
   particular, it does not introduce any signaling between the peers.

2. **Terminology**

   Multi-addressed node
      A mobile or multi-homed node whoose IP address may change.

   Correspondent node
      A multi-addressed node's peer, which may itself be mobile or
      multi-homed.

   Mobility protocol
      A protocol for mobility management executed between a mobile node
      and a correspondent node. Examples are Mobile IPv6 [5] or the
      mobility extensions [3] of the Host Identity Protocol [2].

   Multi-homing protocol
      A protocol for multi-homing management executed between a multi-
      homed node and a correspondent node. An example for a site multi-
      homing protocol is the Level 3 Multihoming Shim protocol [4].

   Binding
      The internal state at the correspondent node tying some form of
      identity of a multi-addressed node to the IP address(es) that the
      multi-addressed node uses at a particular time.

   Binding update
      The process of adding a new IP address to a binding or removing an
      existing IP address from a binding.

   Verified IP address
      An IP address at which the multi-addressed node that claims the
      address has been shown to be reachable.

   Unverified IP address
      An IP address for which no reachability verification has yet been
      accomplished.

   Credit
      The number of bytes that a correspondent node has recently
      received from a multi-addressed node. The correspondent node uses
      the credit to determine how much data it can securely send to an
      unverified IP address of the multi-addressed node.

   Credit counter
      A variable, maintained by the correspondent node, that shows a
      particular multi-addressed node's current credit.

   Credit aging

A function that gradually reduces a multi-addressed node's credit
over time.

Flooding attack
A variant of a denial-of-service attack that is characterized by a
victim being bombarded with unwanted packets at a rate that the
victim, and possibly the victim's access network, cannot handle.

Amplification
The ratio between the data volume that the victim of a flooding
attack is exposed to and the data volume that the attacker itself
generates.

3. Overview

Concurrent reachability verification requires protection against
spoofed unverified IP addresses and redirection-based flooding
attacks. Credit-Based Authorization is a technique that provides
such protection based on the following three hypotheses:

1. A flooding attacker typically seeks to somehow multiply the
    packets it assembles for the purpose of the attack because
    bandwidth is an ample resource for many attractive victims.

2. An attacker can always cause unamplified flooding by generating
    bogus packets itself and sending them to its victim directly.

3. Consequently, the additional effort required to set up a
    redirection-based flooding attack pays off for the attacker only
    if amplification can be obtained this way.

On this basis, rather than eliminating malicious packet redirection
in the first place, Credit-Based Authorization prevents any
amplification that can be reached through it. This is accomplished
by limiting the data a correspondent node can send to an unverified
IP address of a multi-addressed peer by the data that the
correspondent node has recently received from that peer. A
redirection-based flooding attack thus becomes no more attractive
than pure direct flooding, where the attacker itself sends bogus
packets to the victim. It is actually less attractive given that the
attacker needs to maintain a context for mobility or multi-homing
management in order to coordinate the redirection.

```
     multi-addressed node       correspondent node
                  |                         |
                  |                         |
  IP address |--data----------------->| credit += size(data)
    verified |                         |
             |--data----------------->| credit += size(data)
             |<----------------data--| don't change credit
             |                         |
  IP address + IP address changes      |
   unverified |<----------------data--| credit -= size(data)
             |--data----------------->| credit += size(data)
             |<----------------data--| credit -= size(data)
             |                         |
             |<----------------data--| credit -= size(data)
             |                     X credit < size(data) ==> drop!
             |                         |
  IP address |                         |
     verified |<----------------data--| don't change credit
             |                         |
```
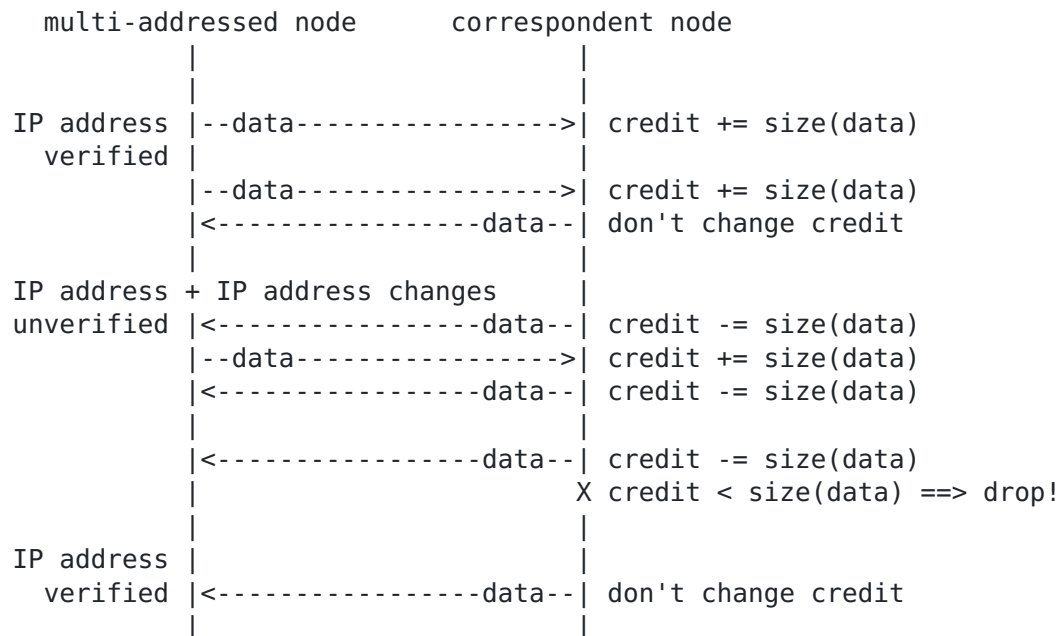
                   Figure 1: Credit Maintenance

Figure 1 illustrates the specifics of Credit-Based Authorization for
an exemplifying exchange of data packets: The correspondent node
measures the bytes received from the multi-addressed node. These
bytes are called the multi-addressed node's "credit" and are kept by
the correspondent node in a "credit counter". When the multi-
addressed node changes IP connectivity and registers a new IP
address, the correspondent node labels this address UNVERIFIED first.
Packets may be sent to an unverified IP address as long as the packet
sizes do not exceed the currently available credit. For each such
packet, the correspondent node reduces the credit counter by the
packet size. The multi-addressed node's reachability at the new IP
address is meanwhile verified. Once the verification concludes, the
correspondent node relabels the new IP address from UNVERIFIED to
VERIFIED. Packets can then be sent to the address without
restrictions. When insufficient credit is left while the IP address
is still in UNVERIFIED state, the correspondent node stops sending
further packets to the address until the verification completes. The
correspondent node may drop these packets or buffer them for later
transmission after the IP address has changed to VERIFIED state.
Figure 1 does not show signaling packets from a mobility or multi-
homing protocol.

The correspondent node ensures that the multi-addressed node's
acquired credit gradually decreases over time. This "credit aging"
prevents the multi-addressed node from building up credit over a long
time. A malicious node with a slow Internet connection could

otherwise provision for a burst of redirected packets which does not relate to its own upstream capacity.

Credit-Based Authorization does not require support from the multi-addressed node and does not introduce any signaling between the peers. A faithful multi-addressed node, communicating with a correspondent node in a typical manner, automatically earns credit for sending packets to the correspondent node. It neither needs to understand that Credit-Based Authorization is effective at the correspondent node, nor does it have to have an idea of how much credit it has at a particular point in time.

4. Detailed Specification

Credit-Based Authorization requires a correspondent node to store the state of each multi-addressed node's IP address(es), maintain a credit counter for each multi-addressed node, and execute an exponential aging function on each credit counter. This is explained in detail in the following.

4.1 IP Address States

Mobility and multi-homing protocols typically require a correspondent node to bind a multi-addressed node's changing IP address to some form of identity of the multi-addressed node. E.g., in Mobile IPv6 [5], this "binding" is between a mobile node's home address and current care-of address. In the mobility extensions [3] of the Host Identity Protocol [2], the multi-addressed node's current locators are bound to a Host Identity Tag. Credit-Based Authorization in addition requires the correspondent node to associate each IP address with a state, VERIFIED or UNVERIFIED, in order to be able to determine whether or not packets sent to the multi-addressed node's current IP address are subject to credit limitations.

When a multi-addressed node changes its point of IP attachment and configures a new IP address, a "binding update" is initiated in order to inform the correspondent node of that new IP address. The new IP address is initially set to UNVERIFIED state at the correspondent node. The multi-addressed node may start to send packets from the new IP address as soon as it has dispatched the signaling message(s) that the mobility or multi-homing protocol provides for the binding update. However, packets that the correspondent node may sent to an IP address in UNVERIFIED state are subject to the limitations specified in Section 4.2.

At some time after the multi-addressed node has sent the signaling
message(s) for the binding update, the mobility or multi-homing
protocol initiates reachability verification for the new IP address
and conveys the result to the correspondent node.

If reachability verification confirms that the multi-addressed node
is present at the new IP address, the correspondent node changes the
state of this address from UNVERIFIED to VERIFIED. Any limits
imposed on packets that the correspondent node sends to the new IP
address do no longer apply as of then.

If the outcome of reachability verification is that the multi-
addressed node is not reachable at the new IP address, the
correspondent node MUST stop sending packets to that IP address.
However, the correspondent node MAY keep the unverified IP address
within the binding and continue to accept packets that the multi-
addressed node sends from the IP address. This is reasonable given
that reachability verification may fail for reasons outside the
influence of the multi-addressed node, e.g., due to packet loss on
the path between the multi-addressed node and the correspondent node.
It is the responsibility of the mobility or multi-homing protocol to
repeat reachability verification an appropriate number of times in
case of failure.

4.2 Handling Payload Packets

A correspondent node maintains a credit counter for each multi-
addressed node it communicates with. All IP addresses within a
binding, both verified and unverified ones, map to the same credit
counter. New credit counters are initialized to zero.

```
   inbound     _____
   packet    |                    |
     |       |     locate peer's  |
    '---->   |      binding and   |
             |     credit counter |
             '_____'
                       |
                       |
                       |
                       v

     _____            _____
    |                    |          |                    |
    |      increase      |          |  deliver packet    |
    |   credit counter   |--------->|   to application   |
    |   by packet size   |          |                    |
    '_____'          '_____'
```
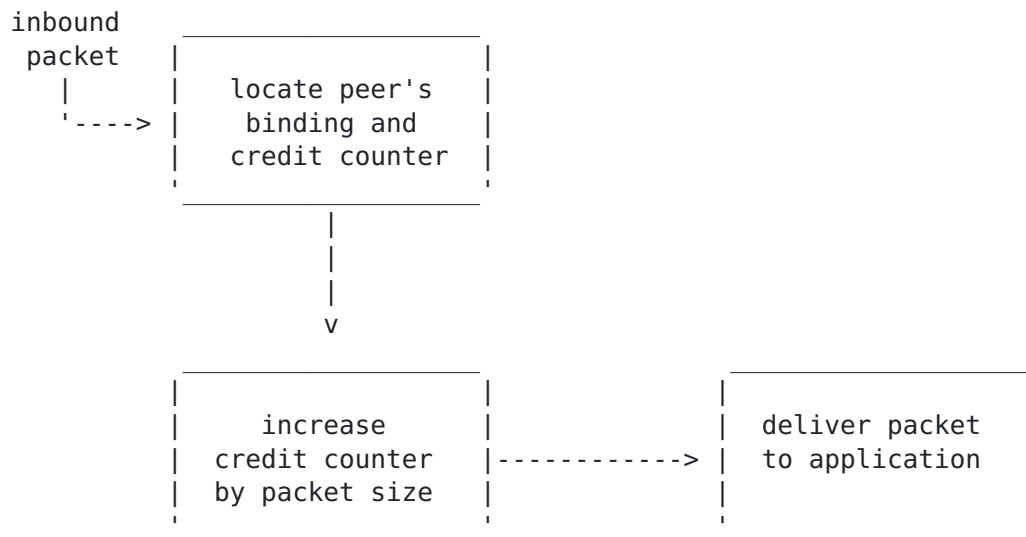
Figure 2: Inbound Packet Handling

   When the correspondent node receives a packet from a multi-addressed
   node (cf. Figure 2), and the packet's IPv6 Source Address is
   currently bound to the multi-addressed node, the correspondent node
   SHOULD increase that multi-addressed node's credit counter by the
   size of the received packet. In particular, it does not matter
   whether the state of the packet's IPv6 Source Address is VERIFIED or
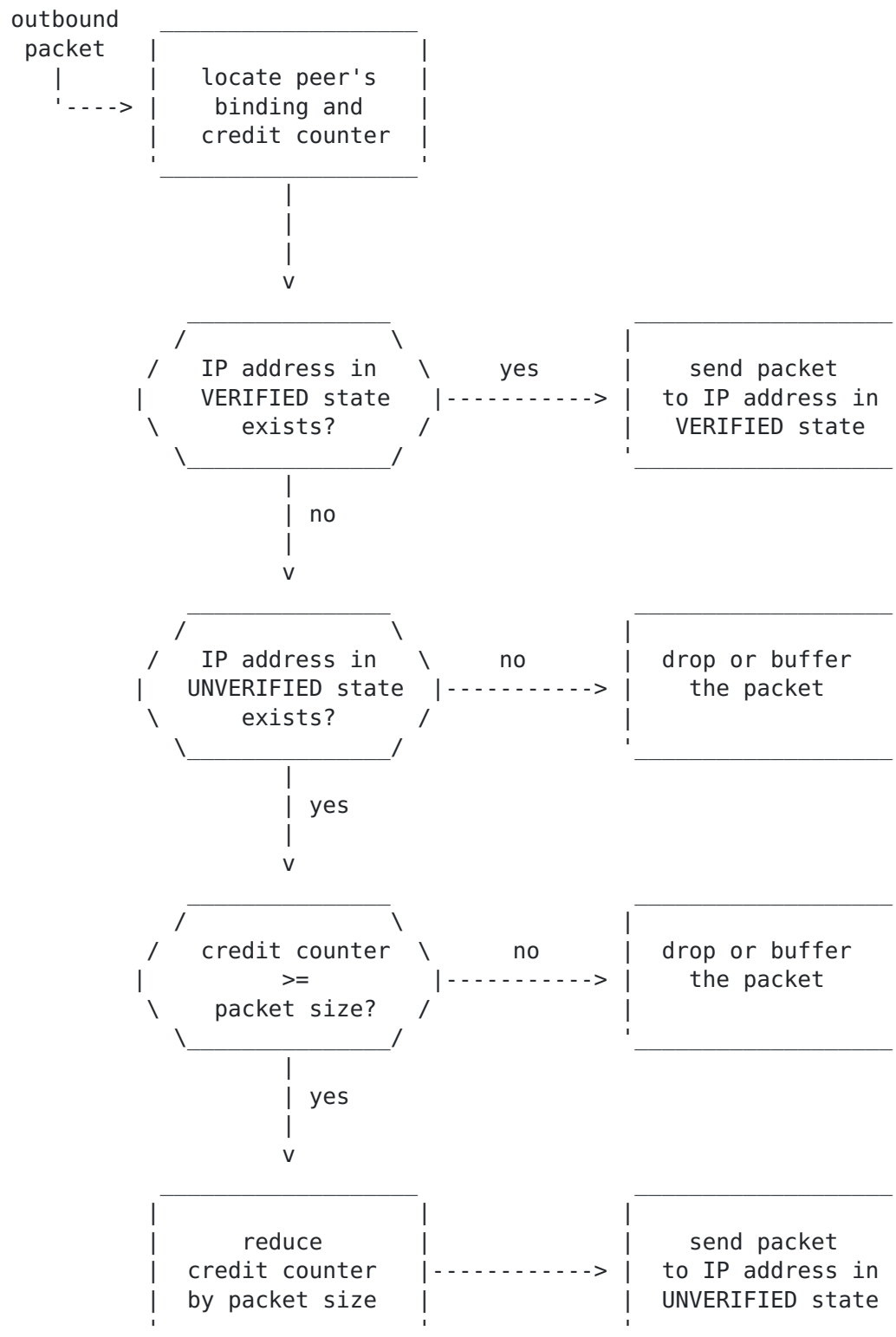   UNVERIFIED.

```
outbound    _____
 packet   |                    |
   |      |     locate peer's  |
   '----> |      binding and   |
          |     credit counter |
          '_____'
                     |
                     |
                     |
                     v

         _____              _____
        /                 \            |                    |
       /   IP address in   \    yes    |    send packet     |
      |    VERIFIED state   |--------->|  to IP address in  |
       \      exists?      /           |    VERIFIED state  |
        _____/            '_____'
                 |
                 | no
                 |
                 v

         _____              _____
        /                 \            |                    |
       /   IP address in   \    no     |   drop or buffer   |
      |   UNVERIFIED state  |--------->|     the packet     |
       \      exists?      /           |                    |
        _____/            '_____'
                 |
                 | yes
                 |
                 v

         _____              _____
        /                 \            |                    |
       /   credit counter  \    no     |   drop or buffer   |
      |         >=          |--------->|     the packet     |
       \     packet size?  /           |                    |
        _____/            '_____'
                 |
                 | yes
                 |
                 v

          _____              _____
        |                  |           |                    |
        |      reduce       |          |    send packet     |
        |   credit counter  |--------->|  to IP address in  |
        |   by packet size  |          |   UNVERIFIED state |
        '_____'           '_____'
```

Figure 3: Outbound Packet Handling

When the correspondent node has a packet for the multi-addressed node
(cf. Figure 3), it SHOULD send the packet to an IP address in
VERIFIED state if such an address exists in the multi-addressed
node's binding. In case no IP address currently bound to the multi-
addressed node is in VERIFIED state, the correspondent node selects
an IP address in UNVERIFIED state provided that such an address is
available. If an unverified IP address exists, the correspondent
node checks to see whether it can send the packet to that address:
In case the value of the credit counter is higher than the size of
the packet or equal to it, the correspondent node reduces the credit
counter by the packet size and sends the packet to the unverified IP
address. If the credit counter is too low, the packet MUST be
discarded or buffered until reachability verification succeeds.
Should the mobility or multi-homing protocol support a stable IP
address via which the multi-addressed node is permanently reachable,
possibly through a sub-optimal routing path, the correspondent node
may also send the packets to that stable IP address until the multi-
addressed node becomes again directly reachable through a verified IP
address. A Mobile IPv6 home address is an example of such a stable
IP address.

4.3 Credit Aging

The correspondent node ensures that all credit counters maintained
for its multi-addressed peers gradually decrease over time. For
this, it multiplies each credit counter with a factor,
CreditAgingFactor, of less than one in fixed time intervals of
CreditAgingInterval length. Such credit aging prevents a malicious
peer with poor uplink capacity from building up credit at a very slow
speed and using this, all at once, for a burst of redirected packets.
At the same time, credit aging naturally limits the rate at which the
correspondent node can durably sent to an IP address in UNVERIFIED
state.

Choosing appropriate values for CreditAgingFactor and
CreditAgingInterval is important to facilitate applications where the
correspondent node sends at a higher rate than the multi-addressed
node. If CreditAgingFactor or CreditAgingInterval are too small, the
credit counter might be too low to allow for packets being sent to an
unverified IP address. The values specified in this document work
well when the host transfers a file to the peer via a TCP connection
and the end-to-end round-trip time does not exeed 500 milliseconds.

[5](). **Security Considerations**

   Essentially three types of flooding attacks are already possible in
   today's Internet: direct attacks, reflection attacks, and
   distributed attacks. The following analysis compares these attack
   types with those that could become possible by the introduction of
   mobility and multi-homing support if no preventive measures are
   taken. The power of a flooding attack is thereby measured by its
   potential for amplification. It is shown that the new attack types
   would facilitate much higher amplification than conventional attack
   types, and that a combination of concurrent reachability verification
   and Credit-Based Authorization can neutralize this disadvantage.

[5.1]() **Conventional Types of Flooding**

   In a direct flooding attack, the attacker simply generates the
   flooding packets and sends them to the victim by itself. Direct
   flooding attacks are an inherent vulnerability of the Internet
   architecture given that the routing infrastructure delivers packets
   independently of whether they have actually been requested by the
   recipient. Firewalls mitigate the issue to some extent in that they
   block undesired traffic at some point close to the end of the routing
   path. The flooding packets are thus screened from a victim node. On
   the other hand, the flooding packets may still exhaust downstream
   capacities of an entire network. Firewalls are generally unsuitable
   to prevent this. Inverse firewalls at the attacker's side screen the
   undesired traffic close to the source, but most likely an attacker
   can choose a location where inverse firewalling is not performed.

   In an indirect reflection attack, the attacker tricks a third node,
   the reflection point, into sending packets to the victim. The
   attacker typically uses a known protocol vulnerability to make the
   reflection point generate these packets [[12]()]. One example is that
   the attacker sends ICMP Echo Request packets, with the IPv6 Source
   Address fields set to the victim's IP address, to the reflection
   point. The reflection point, in turn, sends ICMP Echo Reply packets
   "back" to the victim. Another example is that the attacker sends TCP
   SYN packets, again with false source addresses, to the reflection
   point, which in turn sends TCP SYN-ACK packets to someone who does
   not expect these packets. Since most TCP servers are configured so
   that they resend a TCP SYN packet multiple times when failing to
   receive an acknowledgment, this reflection attack can even produce
   small amplification. As the examples show, reflection attacks are
   generally characterized by the attacker sending packets with spoofed
   IPv6 Source Address fields. The amplification possible through
   reflection is generally rather limited, however, since most protocols
   refrain from sending large amounts of data to an address before some

assurance has been obtained that there is indeed a node willing to
accept packets at the other end.

Distributed flooding attacks provide more significant amplification.
Here, the attacker takes over control of other nodes by compromising
them with viral software, which it typically distributes by
conventional means. The infected nodes are then programmed to
jointly send packets to the victim. Typically, the attack proceeds
automatically, and the attacker itself does not send itself packets
to the victim. Distributed flooding attacks are of a different
quality than the flooding attacks described afore because they
generally exploit implementation vulnerabilities in operating systems
rather than vulnerabilities in standard networking protocols.

5.2 Redirection-Based Flooding

Redirection-based flooding attacks are a fourth attack type, which
mobility and multi-homing protocols could introduce if they fail to
provide appropriate protection. In such an attack, the perpetrator
requests a server to transmit a large file, e.g., a video, and
subsequently misuses a mobility or multi-homing protocol to redirect
this download to the IP address of its victim. The transport
protocol may require the attacker to spoof acknowledgment on behalf
of the victim. But since the acknowledgments are typically smaller
in size and number than the data packets that the server generates,
the amplification in this attack can be much higher than in the
discussed ICMP and TCP flooding attacks.

Today's transport protocols were developed for an Internet where
nodes use stable IP addresses, hence most of them perform
reachability verification, if at all, only at some early stage during
connection establishment. E.g., TCP requires the receiver to obtain
a random 32-bit sequence number during the initial handshake.
Mobility and multi-homing protocols defeat the purpose of such
reachability verification as they introduce the ability to redirect
packets subsequently to the initial handshake. Referring to the
example of the TCP download, the attacker could execute the initial
handshake procedure via its own IP address, and use the sequence
number obtained that way to spoof acknowledgments after it has
redirected the session to the IP address of its victim.

5.3 Protection against Redirection-Based Flooding

Most mobility and multi-homing protocols execute an IP-layer
reachability check as a protection against redirection-based flooding
whenever a node changes its IP address. Given that the delay of such

a check can have a noticable impact on applications, a
straightforward strategy is to execute reachability verification
concurrently while packets are already being sent to the new IP
address. This requires additional protection for the phase during
which packets are sent to the new IP address although the validity of
that IP address is still questionable. Credit-Based Authorization
provides this protection in that it limits the impact of such
redirection to what could also be accomplished---with much less
coordinative effort---through a direct flooding attack.
Specifically, the combination of concurrent reachability verification
and Credit-Based Authorization does not prevent malicious redirection
per se, but it prevents its use from a location off the path towards
the flooded victim as well as any amplification in the quantity of
redirected packets. As a result, a redirection-based flooding attack
becomes as effective as a direct flooding attack.

5.4 Alternatives to Credit-Based Authorization

There are alternatives to Credit-Based Authorization which can
protect against misuse of mobility or multi-homing protocols for
redirection-based flooding attacks. One alternative is to perform
reachability verification in a concurrent way only with nodes from a
trusted community. Mobility and multi-homing protocols usually
authenticate a node during a binding update, providing a way for
secure identification. The correspondent node can thus decide
whether or not to use the new IP address before the result of
reachability verification becomes known. For instance, the
correspondent node may be a corporate server which grants concurrent
reachability verification exlusively to nodes from the corporate
network.

Mobility and multi-homing protocols that use the IPv6 Source Address
field to signal a new IP address to the correspondent node may rely
on ingress filtering [11] to be enforced within the multi-addressed
nodes' networks. Ingress filtering is a function that routers may
execute to prevent packets with spoofed source addresses from leaving
a network. The natural crux with ingress filtering, however, is that
the correspondent node in general does not know whether or not
ingress filtering has been performed on packets received from a
particular multi-addressed node. Furthermore, the granularity of
ingress filtering decreases with the distance between the access
network and the router that executes it. In an optimal case, ingress
filtering is directly applied in the first-hop router. But even then
may an attacker use a false IPv6 Source Address as long as the
network prefix is correct.

A third alternative to Credit-Based Authorization is to identify and

blacklist malicious nodes based on their observed behavior. Credit-
Based Authorization is a proactive strategy, whereas behavior-based
blacklisting is a reactive one. The advantage of a reactive approach
is that a faithful multi-addressed node does not need to invest
effort (i.e., collect credit) in order to be able to receive packets
at an unverified IP address. This can accommodate a multi-addressed
node having trouble to collect sufficient credit---be it because its
IP address changes too frequently, or because its application
generates too little data. On the other hand, a reactive approach by
definition fails to prevent misuse of concurrent reachability
verification in advance. What it can do is contain the damage of
such misuse and punish the perpetrator. Punishment depends on an
administrative relationship between the multi-addressed node and the
correspondent node, however, and such a relationship may not exist.
finally, behavior-based blacklisting requires a heuristic to
determine what behavior is considered "ill". Yet choosing the right
heuristic is far from trivial. An inappropriate heuristic may punish
a faithful mobile node or overlook an evil one.

5.5 Alternatives to Credit Aging

Credit-Based Authorization uses exponential aging to slowly reduce
collected credit over time. An alternative to exponential aging
would be a constant upper bound on the credit. However, such a limit
would allow an attacker to acquire credit at a very slow speed,
possibly through a slow Internet connection, and to use this credit
quickly for a burst of packets redirected to a victim. Collected
credit may also be kept for a long time before it is eventually used.
This would allow the attacker to build up credit at multiple
correspondent nodes one after another. Finally, a constant limit
would be insensitive to throughput conditions on the path between the
multi-addressed node and the correspondent node. A limit for a low-
bandwidth connection is certainly inappropriate for a high-bandwidth
connection, and vice versa. Exponential aging does not have these
disadvantages and was thus considered a more appropriate approach
than limiting a mobile node's credit by a constant upper bound.

6. Protocol Constants

```
        CreditAgingFactor        7/8
        CreditAgingInterval      5 seconds
```

7. Acknowledgment

The necessity for a mechanism to prevent or discourage misuse of
concurrent reachability verification for amplified redirection-based
flooding attacks was sparked by a fruitful discussion on the MIP6 and
MOBOPTS mailing lists. Credit-Based Authorization was developed as a
candidate solution, and a first presentation was given at the 59th
IETF meeting in Seoul, Republic of Korea. For their interest and
valuable feedback, the authors thank the MIP6 and MOBOPTS
communities, in particular Roland Bless, Mark Doll, Francis Dupont,
Gregory Daley, Lars Eggert, James Kempf, Rajeev Koodli, Tobias
Kuefner, Marco Liebsch, Gabriel Montenegro, Nick (Sharkey) Moore,
Pekka Nikander, Erik Nordmark, Charles Perkins, and Kilian Weniger
(listed in alphabetical order). Thanks are also due to the
development team of the Kame-Shisa Mobile IPv6 implementation.

8. References

8.1 Normative References

[1]   Vogt, C., "Early Binding Updates for Mobile IPv6",
      draft-vogt-mobopts-early-binding-updates (work in progress)
      (work in progress).

8.2 Informative References

[2]   Moskowitz, R., Nikander, P., Jokela, P., and T. R., "Host
      Identity Protocol", IETF Internet Draft
      draft-ietf-hip-base-04.txt (work in progress), October 2005.

[3]   Henderson, T., "End-Host Mobility and Multihoming with the Host
      Identity Protocol", IETF Internet Draft
      draft-ietf-hip-mm-02.txt (work in progress), July 2005.

[4]   Bagnulo, M. and E. Nordmark, "Level 3 multihoming shim
      protocol", IETF Internet Draft draft-ietf-shim6-proto-03.txt
      (work in progress), December 2005.

[5]   Johnson, D., Perkins, C., and J. Arkko, "Mobility Support in
      IPv6", RFC 3775, June 2004.

[6]   Nikander, P., Arkko, J., Aura, T., Montenegro, G., and E.
      Nordmark, "Mobile IP Version 6 Route Optimization Security
      Design Background", IETF Request for Comments 4225,
      December 2005.

   [7]   Vogt, C. and M. Doll, "Efficient End-to-End Mobility Support in
          IPv6", To appear in the proceedings of the IEEE Wireless
          Communications and Networking Conference, IEEE, April 2006.

   [8]   Nikander, P., Arkko, J., Aura, T., Montenegro, G., and E.
          Nordmark, "Mobile IP version 6 Route Optimization Security
          Design Background", draft-ietf-mip6-ro-sec (work in progress)
          (work in progress).

   [9]   Aura, T., Roe, M., and J. Arkko, "Security of Internet Location
          Management", In Proceedings of the 18th Annual Computer
          Security Applications Conference, Las Vegas, Nevada, USA.,
          December 2002.

   [10]   Arkko, J. and C. Vogt, "Credit-Based Authorization for Binding
          Lifetime Extension",
          draft-arkko-mipv6-binding-lifetime-extension (work in progress)
          (work in progress).

   [11]   Ferguson, P. and D. Senie, "Network Ingress Filtering:
          Defeating Denial of Service Attacks which employ IP Source
          Address Spoofing", RFC 2827, May 2000.

   [12]   Paxson, V., "An Analysis of Using Reflectors for Distributed
          Denial-of-Service Attacks", Computer Communication Review
          31(3)., July 2001.

   [13]   Anderson, R., "Why Information Security is Hard -- An Economic
          Perspective", In Proceedings of the 17th Annual Computer
          Security Applications Conference, New Orleans, Louisiana, USA.,
          December 2001.

   [14]   Perkins, C., "Precomputable Binding Management Key Kbm for
          Mobile IPv6", draft-ietf-mip6-precfgkbm (work in progress)
          (work in progress).

Authors' Addresses

   Christian Vogt
   Institute of Telematics
   Universitaet Karlsruhe (TH)
   P.O. Box 6980
   76128 Karlsruhe
   Germany

   Email: chvogt@tm.uka.de

Jari Arkko
Ericsson Research NomadicLab
FI-02420 Jorvas
Finland

Email: jari.arkko@ericsson.com

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment