Credit-Based Authorization

⟨draft-vogt-mipv6-credit-based-authorization⟩

Christian Vogt, chvogt@tm.uka.de

Jari Arkko, jari.arkko@nomadiclab.com

Roland Bless, bless@tm.uka.de

Mark Doll, doll@tm.uka.de

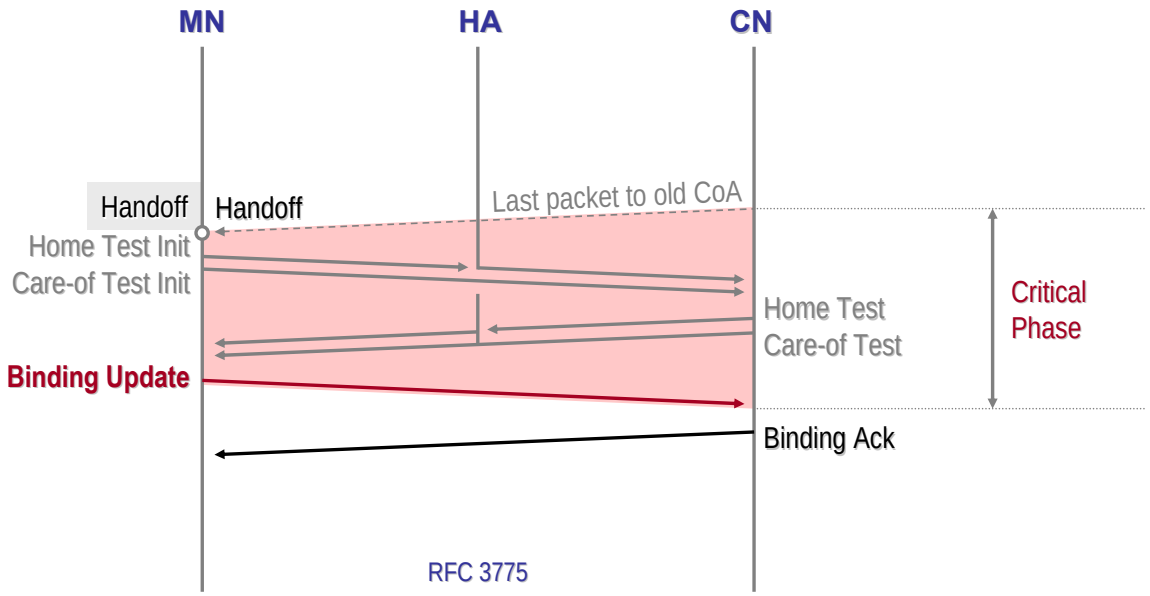
Tobias Küfner, kuefner@tm.uka.de

Mobopts Meeting, 60th IETF Meeting, San Diego

August 2, 2004

This document contains annotations for each presentation slide.

Mobile IPv6 Correspondent Registrations

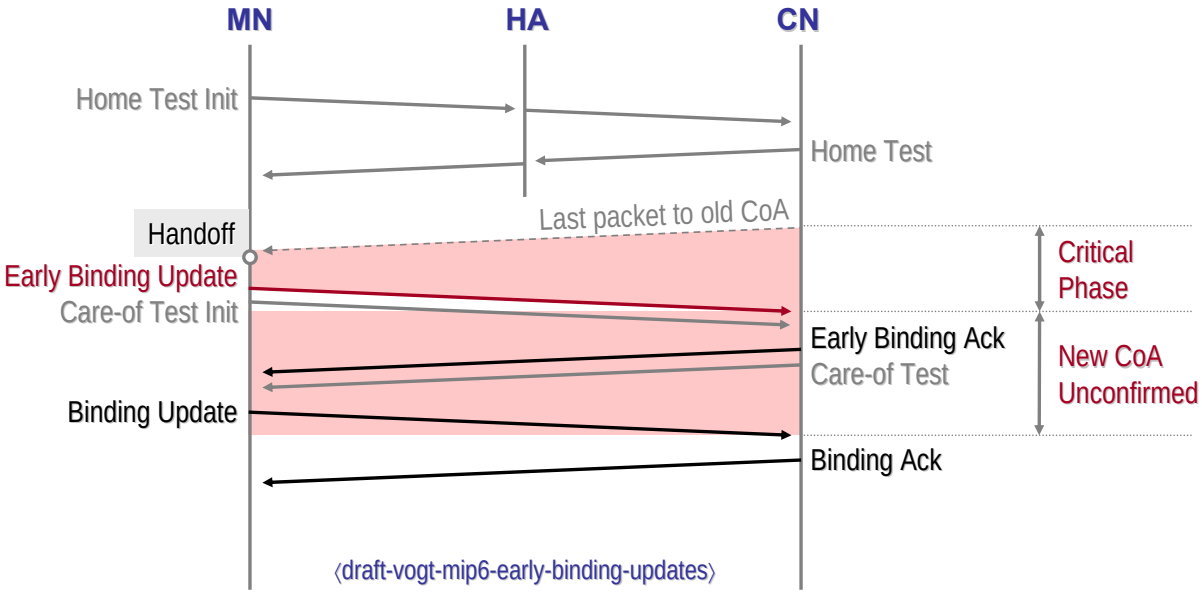


Mobile IPv6 Correspondent Registrations, Annotations

This is an illustration of a standard Mobile IPv6 *correspondent registration*.

A correspondent registration consist of a return-routability procedure and the registration proper. The return-routability procedure is a home-address test and a care-of-address test. If performed in parallel, these tests take at least a round-trip time. The registration proper takes another round-trip time such that the *critical time,* during which two peers cannot fully communicate, is at least two round-trip times.

Early Binding Updates



Early Binding Updates, Annotations

At the 59th IETF meeting in Seoul, we have seen that Early Binding Updates [1] can reduce the latency of a correspondent registration to only one round-trip time.

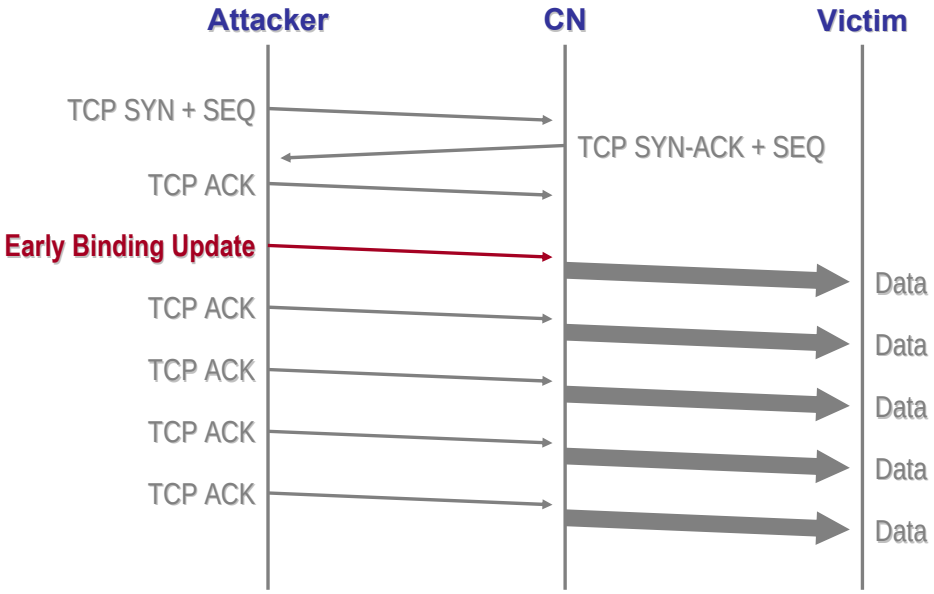
Early Binding Updates eliminate the latency of the return-routability procedure in the following way: The home-address test is *proactively* performed before a handover. The care-of-address test takes place *concurrently* with data transfer to and from a new care-of address. Hence, what remains to be done during the critical phase is the registration proper.

There was a long discussion on the mailing list whether the concurrent care-of-address test introduces new security threats. Indeed, we have a new phase of uncertainty, during which the correspondent node doesn't yet know whether or not the mobile node is actually present at the new care-of address. Let's call the new care-of address *unconfirmed* during this phase, and let's call it *confirmed* thereafter.

Why can an unconfirmed care-of address be an issue? Consider the following example…

[1] Christian Vogt, Roland Bless, Mark Doll, Tobias Küfner: "Early Binding Updates for Mobile IPv6", draft-vogt-mip6-early-binding-updates

Amplified Flooding



Amplified Flooding, Annotations

In this scenario, an attacker intends to cause denial of service at a victim by flooding this victim with unwanted data.

For this, the attacker requests an arbitrary content server to send to it a large file. The two peers perform a standard TCP handshake. Once data is flowing, the attacker claims that it has moved, and it tells the content server to redirect the data to a new care-of address. In fact, the attacker uses its victim's IP address as the new care-of address. While the new care-of address is unconfirmed, the server will thus send all data to the victim.

The attacker can keep the data flow alive, and even *accelerate* it, by sending spoofed TCP acknowledgements. After all, the attacker has learned the initial sequence number during the TCP handshake. A limit on an unconfirmed care-of address's lifetime does not help, because the attacker can simply resend the Early Binding Update message, refreshing the unconfirmed care-of address again and again.

What is worse, the attacker can repeat the same procedure with any number of correspondent nodes.

The severity of this flooding attack is that its *amplified*. The victim sees itself bombarded with back-to-back full-size TCP segments, whereas the attacker only sporadically generates a small acknowledgement.

It is true that flooding attacks do exist already today. However, to gain a comparable amplification factor, an attacker would have to get control over a high number of other nodes—for instance through viral-code distribution—that assist it during the attack. In contrast, in the attack shown here, the attacker can use any correspondent node providing mobility support.

Obviously, something needs to be done to prevent misuse of unconfirmed care-of addresses for amplified flooding attacks. Credit-Based Authorization is an idea to do this.

Credit-Based Authorization

- **CN** maintains **CREDIT** per MN
- CN receives packet from MN:

  CREDIT += packet_size
- CN sends packet to MN @ unconfirmed CoA:

  CREDIT −= packet_size • X

  X > 1.0
- **CREDIT ≈ 0 ?**

  Send packet to HoA
- Volume **sent** to MN @ unconfirmed CoA **<** Volume **received** from MN

Credit-Based Authorization, Annotations

Credit-Based Authorization [2] weighs up the data volume that the correspondent node sends to a mobile node's unconfirmed care-of address and the data volume that this mobile node has previously sent to the correspondent node.

More specifically, the correspondent node maintains a *credit account* for each mobile node it communicates with. When the correspondent node receives a packet from a particular mobile node, the correspondent node gives the mobile node a credit amount equal to the size of the received packet.

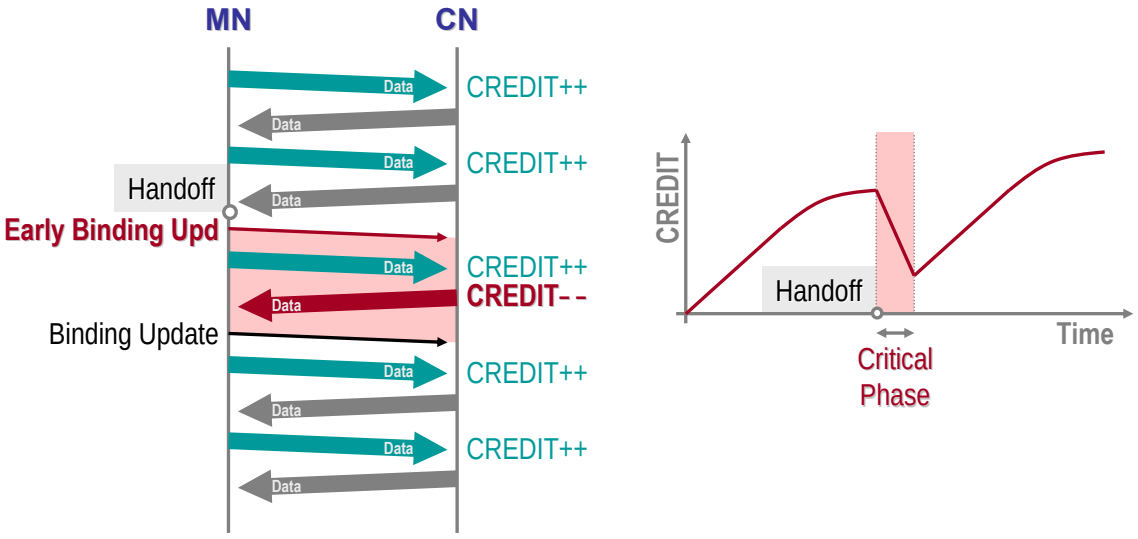
When the correspondent node sends a packet to the mobile node, if the mobile node's care-of address is unconfirmed, the correspondent node charges a credit amount proportional to the size of the to-be-sent packet. A protocol-configuration variable, here denoted as X, ensures that the mobile node's credit shrinks faster than it grows.

What happens when the mobile node's care-of address is unconfirmed and no credit is left? In this case, the correspondent node sends packets to the mobile node's home address instead of sending them to the care-of address. Recall that, with Early Binding Updates, the home address is authenticated at all times. Sending packets to the home address is therefore legitimate, regardless of whether the care-of address is unconfirmed or confirmed. The correspondent node will continue to accept packets that the mobile node sends from its care-of address during this time.

When the mobile node's care-of address becomes confirmed, packets for the mobile node will again be sent to the care-of address.

[2] Christian Vogt, Jari Arkko, Roland Bless, Mark Doll, Tobias Küfner: "Credit-Based Authorization for Mobile IPv6 Early Binding Updates", draft-vogt-mipv6-credit-based-authorization

Credit-Based Authorization (2)



Amplified flooding impossible.
What if traffic is asymmetric?

Credit-Based Authorization (2), Annotations

This is an illustration of Credit-Based Authorization. On the left-hand side, you see a packet exchange between the mobile node and the correspondent node. On the right-hand side, you see how much credit the mobile node has at a particular time during this packet exchange.

When the mobile node sends a packet to the correspondent node, the correspondent node gives the mobile node credit. The packet coming back from the correspondent node does not change the credit because, at this time, the mobile node's care-of address is confirmed. Again, the mobile node sends a packet, and it gets credit. And the correspondent node returns a packet…

You can see on the right-hand side that the mobile node's credit ages over time. This way, we prevent a mobile node from keeping credit indefinitely.

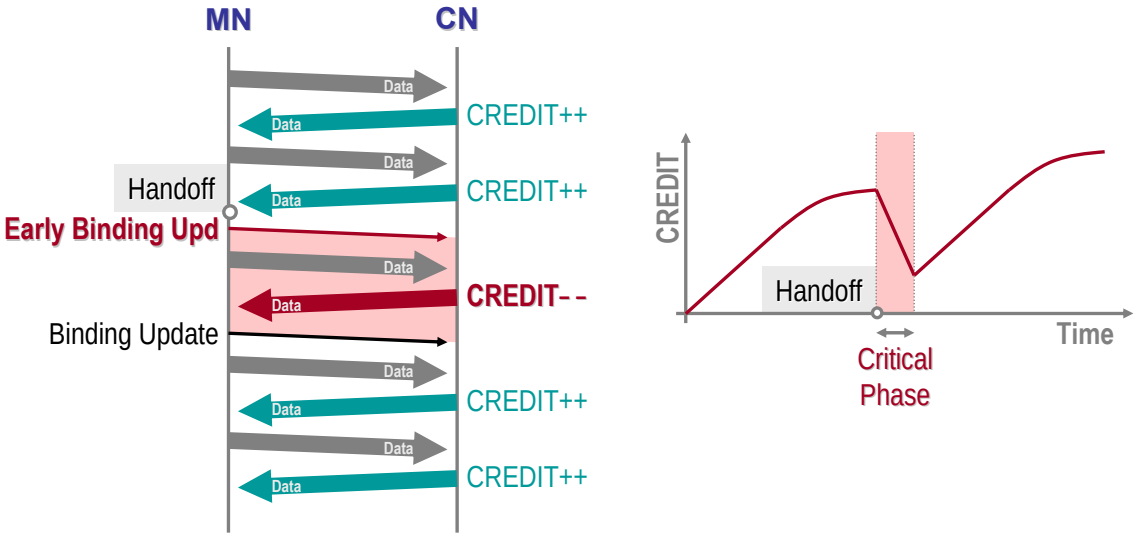
At this time, the mobile node changes its point of network attachment. The new care-of address is now unconfirmed. When the mobile node sends a packet to the correspondent node, it still gets credit. But when the correspondent node sends a packet to the mobile node, some credit will be charged. As you can see on the right-hand side, the credit shrinks faster than it grows.

When the mobile node's care-of address becomes confirmed again, credit will no longer be charged. It still ages, though.

To summarize, the correspondent node will never send more data to an unconfirmed care-of address than it has previously received from the mobile node. This eliminates the threat of amplified flooding attacks. Indeed, since credit shrinks faster than it grows, even non-amplified flooding attacks are discouraged.

One question though is what happens when the correspondent node sends much more data to the mobile node than vice versa. Consider, for example, a streaming application. With this variant of Credit-Based Authorization, credit is given based on the data that the mobile node sends, and it is consumed based on the data that the correspondent node sends. The mobile node might therefore have a hard time earning the amount of credit that it needs for a handover.

CBA Variant 2



Works well with any kind of traffic.
What if packets are lost?

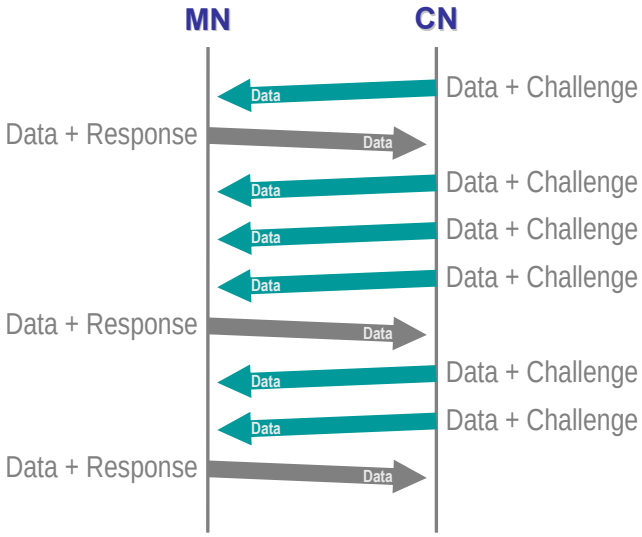
CBA Variant 2, Annotations

A second variant of Credit-Based Authorization behaves more adaptive to applications with asymmetric traffic patterns. It also applies to applications with symmetric traffic patterns.

In the second variant, a mobile node gets credit for packets that it receives—or is expected to receive—from the correspondent node rather than for packets that it sends to the correspondent node. The rationale for this is that a mobile node spends comparable bandwidth, processing power, and memory for both sending and receiving packets.

One question remains: How can the correspondent node determine if the mobile node actually receives the packets that it sends? After all, some packets may get lost along the way. In an extreme situation, a malicious node may hide behind a "bottleneck link", receiving only a fraction of the packets, but collecting credit for all of them. There needs to be a feedback mechanism that provides the correspondent node with the information how many packets get through to the mobile node.

Care-of-Address Spot Checks (for CBA Variant 2)



$$\text{CREDIT} \sim \frac{\text{\# Challenges returned}}{\text{\# Challenges sent}}$$

Care-of-Address Spot Checks (for CBA Variant 2), Annotations

Care-of-address spot checks are a probabilistic mechanism that the correspondent node can use to approximate the delivery ratio of packets it sends to the mobile node.

A care-of-address spot check is a random challenge periodically attached to a packet sent to the mobile node. When the mobile node receives a packet with an attached challenge, it can extract the challenge from the packet and return it with the next packet that it sends to the correspondent node.

The correspondent node can then compare the number of challenges returned to the number of challenges sent. It uses the ratio of these two values as an approximation for the real packet-delivery ratio. When the mobile node gets credit the next time, the amount of credit will be proportional to the approximated ratio.

Analysis: Security

- CN sends **less** to unconfirmed CoA **than** received from (or sent to) confirmed CoA

  ⇒ **No** flooding **amplification**

- CREDIT shrinks faster than grows

  ⇒ **Non-amplified** flooding **unattractive**

Analysis: Security, Annotations

What level of security does Credit-Based Authorization provide?

Credit-Based Authorization ensures that the data volume the correspondent node sends to an unconfirmed care-of address of a mobile node cannot exceed the data volume that the correspondent node has recently received from that mobile node. Or, if we use variant 2 of Credit-Based Authorization, it ensures that the data volume the correspondent node sends to the mobile node's unconfirmed care-of address cannot exceed the data volume that the correspondent node has recently sent to a confirmed care-of address of that mobile node.

Since the mobile node's credit shrinks faster than it grows, the data volume sent to an unconfirmed care-of address is actually much less than the data volume recently received or sent, respectively.

This, in turn, means that amplified flooding attacks become impossible. In addition, non-amplified flooding attacks become very unattractive because of two reasons: First, the data volume that an attacker can make the correspondent node redirect to a victim is much less than the attacker has recently sent or received itself. The attacker would be more efficient if it sent packets to the victim directly. Second, redirected packets bear the attacker's home address in their routing headers. Note that this home address has been verified before through the proactive home-address test.

In summary, with Credit-Based Authorization, amplified flooding become impossible, and non-amplified flooding becomes unattractive.

Analysis: Efficiency

- Use new CoA **while** doing the CoA test

  ⇒ Reduce critical phase by **50%**

- **Transparent** to MN (unless Spot Checks are used)

- Increased **complexity** at CN

- Applicable to other return-routability optimizations

Analysis: Efficiency, Annotations

What benefits in terms of efficiency do we get from Credit-Based Authorization?

With Credit-Based Authorization, we can savely apply Early Binding Updates. In other words, we can savely use a new care-of address while the concurrent care-of-address test is in progress. Together with a proactive home-address test, we can save the latency of the entire return-routability procedure. The critical phase, during which we cannot use a new care-of address, thus reduces from two round-trip times to one round-trip time.

Credit-Based Authorization not only applies to Early Binding Updates. It can be useful for any return-routability optimization that does not omit the care-of-address test. CGA-OMIPv6, or a combination of pre-configured binding-management keys with care-of-address tests, are two examples where Credit-Based Authorization could yield further performance gains.

Open Issues

- What **level of security** is desired?
  - Variant 1 vs. variant 2
  - Variant 2 with vs. w/o Spot Checks
  - Protocol Parameters (X, Aging)
- How **complex** is a CBA implementation?

Open Issues, Annotations

Credit-Based Authorization is still a topic of research, and there are open issues.

One important question is what level of security we want. A related question is whether some of the presented options of Credit-Based Authorization should be abandoned.

For instance, one could argue that credit should only be given for packets that the mobile node sends. This would leave us with variant 1 of Credit-Based Authorization. It might, as mentioned, lead to problems with asymmetric applications.

Variant 2 of Credit-Based Authorization works fine with symmetric and asymmetric applications, because credit allocation and credit consumption is both based on packets flowing into the same direction, i.e., towards the mobile node. Whether the credit grows or whether it shrinks depends on the mobile node's care-of address being confirmed or unconfirmed.

An issue with variant 2 is that the correspondent node cannot see if packets actually reach the mobile node. Care-of-address spot checks can help. They are most probably not even difficult to implement. Yet, they require support from the mobile node, whereas the beauty of Credit-Based Authorization per se is that it is transparent to the mobile node.

Then there is the question of how complex Credit-Based Authorization is to implement. We strongly believe that the additional overhead at the correspondent node is reasonable. And unless we use care-of-address spot checks, Credit-Based Authorization requires no additional signaling. As a proof of concept, we are currently in the process of implementing Credit-Based Authorization. A very basic implementation of Early Binding Updates, without Credit-Based Authorization, is already finished. The implementation will yield practical insight into the efficiency-vs.-overhead tradeoff of Credit-Based Authorization.