Yoid Identification Protocol (YIDP) Specification

Paul Francis

ACIRI
francis@aciri.org

April 2, 2000

Contents

1 Changes

April 2, 2000 Updated for name change from Yallcast to Yoid.

2 Introduction

The Yoid Identification Protocol (YIDP) serves two primary purposes. First, it identifies the yoid group of a given frame. Second, it identifies the immediate source and destination group members of a given frame. By immediate we mean the member that last transmitted the frame and the member that will next receive the frame (as opposed to the original source and ultimate destination). These members are also refered to as the HxH (hop by hop) source and destination members.

YIDP makes some attempt at being able to identify members without the use of IP addresses. The reason is that IP addresses may change over time (for instance a dial-up host that loses a connection and redials), or may undergo

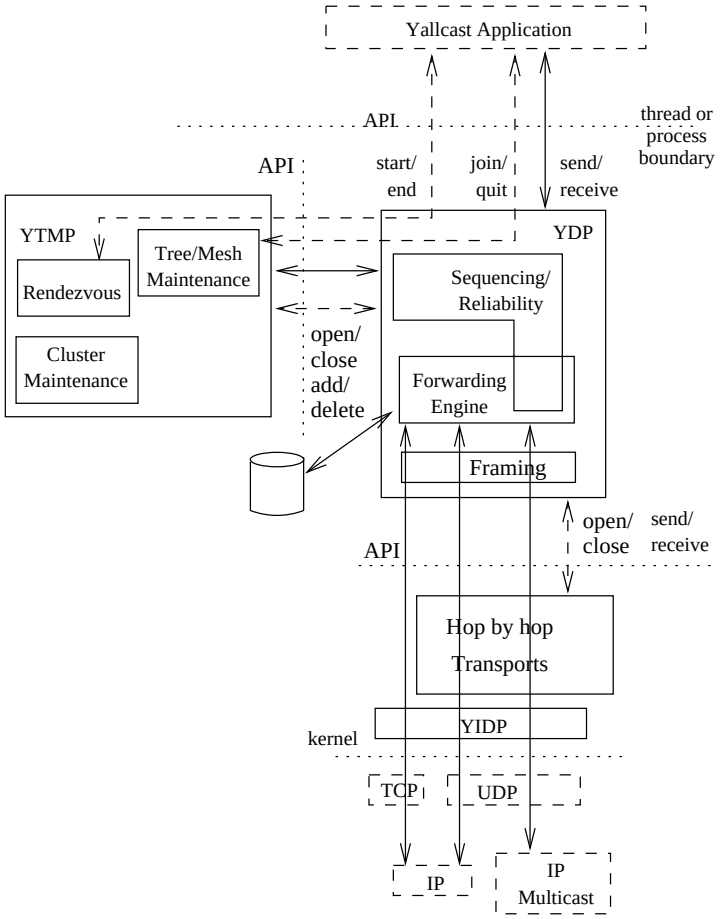


Figure 1: Position of YIDP

translation. Its use with NAT boxes remains problematic because of the general inability to establish incoming connections through NAT boxes.

YIDP and its role in the Yoid architecture is described in broad terms in "Yoid: Extending the Internet Multicast Architecture". This specification assumes the reader is familier with that document.

2.1 Document Status

This spec is very much a snapshot of a work in progress. In particular, we can expect YIDP to evolve considerably as we try it in different environments and find what works and what doesn't.

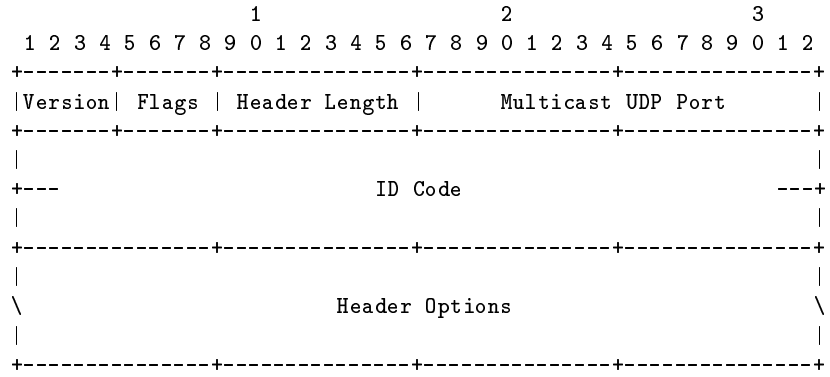
Virtually all of what is in this spec has been implemented, though not yet very thoroughly tested.

3 Position of YIDP

The position of YIDP, relative to other protocols as it might appear in a typical implementation, is shown in Figure 1. YIDP is the first yoid protocol encountered by a received packet. It always runs over either TCP or UDP, and under one of the HxH transport protocols if one exists, or directly under YDP if not.

4 Yoid ID Protocol (YIDP) Header Format

This section describes the header format for the Yoid ID Protocol (YIDP). ZZZZ It runs directly over TCP and UDP. It is formatted as follows:



The fields are defined as follows:

Version: Header version number. Set at 1 for this header.

Flags: The following flags are defined:

Reserved (5-8): Reserved. Transmit as 0, ingore upon reception.

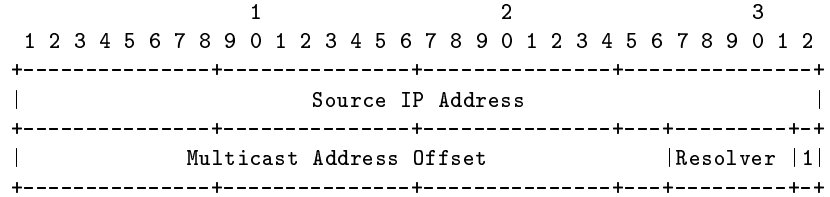
Header Length: The length, in 32-bit words, of the YIDP header including options.

Multicast UDP Port: If the ID Code is for a frame being transmitted over IP unicast (i.e., not for a cluster), then this field is set to 0. Otherwise, this is the UDP Port over which packets for the cluster's IP group are being received (that is, the Destination UDP Port Number).

ID Code: The ID Code is a tag representing the source host name, destination host name or multicast group, and Group ID of this frame. If it is transmitted over unicast IP, then it is assigned by the receiver of the frame. If it is transmitted over multicast IP, then it is assigned by the transmitter of the frame according to specific rules.

If the ID Code is for a frame being transmitted over IP unicast, then an ID Code of all zeros means that the ID Code is unknown, and is what is initially transmitted by a source host before it knows the value that the destination host will assign. Otherwise, the assigned value of the ID Code is at the complete discretion of the destination host.

If the ID Code is for a frame being transmitted over IP multicast (i.e., for a cluster), the assigned value of the ID Code is almost completely determistic, and has significant internal structure, as shown in the following fields:



The fields are defined as follows:

Source IP Address: This is the source IP address of the transmitting host, for the interface over which the packet was transmitted.

Multicast Address Offset: This field is the offset into the range of 2^{26} IP multicast addresses that can be used for clusters. These are the global scope addresses of IP. The offset itself is derived from a hash of the Group ID (as defined in "Yoid Topology Management Protocol (YTMP) Specification"), which in turn is used to assign the IP multicast address. The offset can be derived from the IP multicast address by subtracting 3,892,314,112 from the IP multicast address itself (here treating the IP multicast address as an unsigned integer).

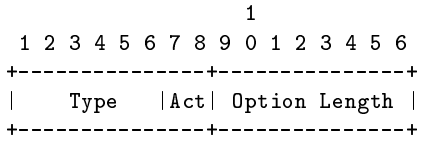
Resolver: This is the only field of the ID Code that is dynamically assigned. It is used to resolve the case where multiple groups have the same IP multicast address and UDP port. It allows up to 31 separate groups with the same multicast address and UDP Port. (In the extremely unlikely case that there are more than 31 such groups, any remaining groups must use the Full ID option with every frame.)

The value 0 is reserved to mean that the ID Code has not yet been resolved. It is used by a host that does not yet know the complete ID Code. The remaining values are assigned dynamically, and generally remain stable over the lifetime of a given cluster.

Note that the only difference between ID Codes transmitted by different hosts in the same group and cluster is the Source IP Address. The rest of the ID Code is unique for a given group. This allows a host on a cluster to be able to identify the group without regard to the source of the frame.

4.1 Header Options

Each header option contains the two-byte fixed part shown below:



Type: This defines the type of option.

Act: This indicates what action should be taken if the option is not recognized, as follows:

Ignore Silently(0): Ignore the option, but continue processing other options and process the frame. Don't notify the sender.

Ignore With Notification (1): Ignore the option, but notify the sender that the option was not recognized with the ASCMP Error Message of type Type Unrecognized.

Drop Silently (2): Drop the entire frame, don't notify the sender.

Drop With Notification (3): Drop the entire frame, and notify the sender.

Option Length: The length of the option, in units of 32-bit words, including the fixed part.

The following sections describe the available header options.

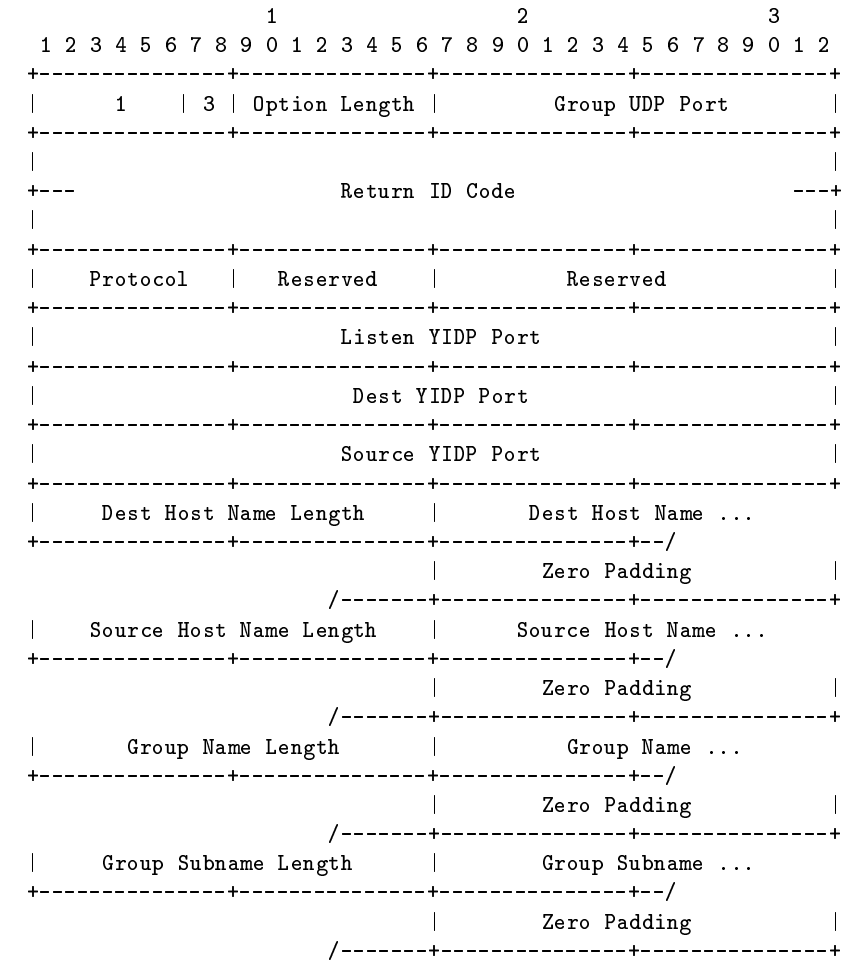
4.1.1 Full ID Option

This option contains the full source host name, destination host name, and Group ID for the frame, where the Group ID consists of the Group Name, the Group Subname, and the Group UDP Port. Its format is as shown below:

These fields are defined as follows:

Fixed Part: The Type code is 1, and the Action is 3 (drop with notification). The Option Length is variable.

Group UDP Port: This is the UDP port number for the group.



Return ID Code: This is the code assigned by the source host (the host transmitting this frame) for the ID Code that should be used in the reverse direction. In other words, a frame transmitted from this frame's destination host to this frame's source host should contain this ID Code in the fixed part of the header. The value 256 is reserved to mean that no Return ID Code is being transmitted.

Protocol: This identifies the next higher layer protocol, as follows:

YDP (1): The next protocol is YDP.

yTCP (2): The next protocol is Yoid TCP (yTCP). With the exception that the checksum is disabled, this runs identically to, and uses the same header format as TCP. It has been fully implemented.

RTP (3): The next protocol is RTP.

yMTP (4): The next protocol is the Yoid Multicast Transport. Its purpose will be to provide reliable data transmission and flow control over a cluster. It has not yet been defined.

yMRTP (5): The next protocol is Yoid Multicast RTP. Its purpose will be to provide flow control without reliability over a cluster. It has not yet been defined.

0, 6 - 255: Currently undefined.

Reserved: Transmit as 0, ingore upon receipt.

Listen YIDP Port: The YIDP port for which the member is listening for frames from new members. In other words, when a remote member wants to initialize transmission of frames to the member, the Dest YIDP Port must contain the value advertised in the Listen YIDP Port. Listen YIDP Port does not change in value for a given member. They are learned via advertisement through the ASCMP protocol. Different members using

the same Host Name and UDP port must have different Listen YIDP Port values. (Indeed, the whole purpose behind this field is to allow multiple members to use the same Host Name and UDP port.)

Dest YIDP Port: The YIDP port identifying the destination member. The Dest YIDP Port will have The value 0 is reserved for future use. The value 1 is reserved to mean that the true Dest YIDP Port of the destination is unknown. All other values are valid Dest YIDP Port values.

If a member is initializing communications with a remote member, then the value of Dest YIDP Port must be what that remote member advertised as its Listen YIDP Port. If a remote member initialized communications, then the value of Dest YIDP Port must be that of the received Source YIDP Port.

Source YIDP Port: The YIDP port identifying the source member. The value 0 is reserved for future use. The value 1 is invalid as a Source YIDP Port. All other values are valid Source YIDP Port values.

If a member is initializing communications with a remote member, then the value of Source YIDP Port is locally chosen. It should be a value that has not been recently used with the remote member, so as to insure that the remote member has flushed any previous association using the same value. If a remote member initialized communications, then the value of Source YIDP Port must be that of the received Dest YIDP Port (which in turn is the value advertised as the Listen YIDP Port).

Dest Host Name Length: The length, in bytes, of the Destination Host Name (not including padding). For unicast IP frames, the destination host is the HxH host receiving the frame. For multicast IP frames, the destination host is NULL, and this field must be set to 0.

Dest Host Name: The DNS name of the destination host. It is padded out, if necessary, to an integral 32-bit boundary by zeros. In the case of a NULL Dest Host Name, this field consists of only two bytes of padding.

Source Host Name Length: The length, in bytes, of the Source Host Name (not including padding). The source host is the HxH host sending the frame.

Source Host Name: The DNS name of the source host. It is padded out, if necessary, to an integral 32-bit boundary by zeros.

Group Name Length: The length, in bytes, of the Group Name (not including padding).

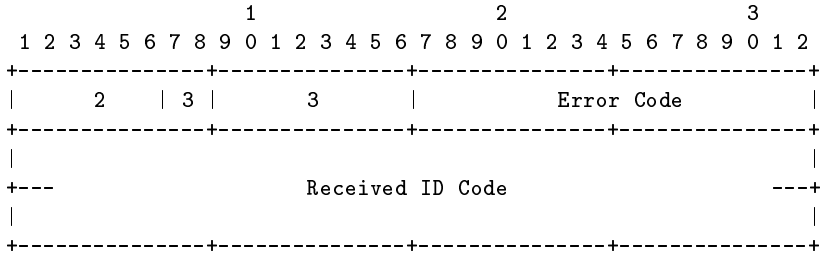
Group Name: The name of this group. The name must be a working DNS name. It is padded out, if necessary, to an integral 32-bit boundary by zeros.

Group Subname Length: The length, in bytes, of the Group Subname (not including padding).

Group Subname: The Subname string for this group, subject to the same character constraints of DNS names. It is padded out, if necessary, to an integral 32-bit boundary by zeros. The Group Subname must be unique among the groups with the same Group Name, but need not be unique globally. It is insensitive to case.

4.1.2 ID Code Error Option

This option is used to convey that a received ID Code is in error. Its format is as shown below:



These fields are defined as follows:

Fixed Part: The Type code is 2, and the Action is 3 (drop with notification). The Option Length is 3.

Error Code: This indicates the reason for the error, as follows:

ID Code Unknown (1): The received ID Code is unknown.

All other values are invalid and must be ignored upon receipt.

Received ID Code: This is the ID Code in the YIDP header that generated this ID Code Error.

5 Yoid ID Protocol (YIDP) Algorithms

5.1 Assignment of the ID Code for IP Unicast Frames

The ID Code is defined by the receiver of the ID Code in the case of IP unicast frames. In the case of IP multicast frames (i.e., frames used with clusters), the ID Code is partly defined by the sender, and partly by the cluster head. This section describes the assignment of the ID Code for IP unicast. The following section describes its assignment for IP multicast.

The ID Code represents uni-directional frame transmission only. ID Codes are assigned seperately be the respective receivers of frames. That is, the destination host of a given frame is the assigner of the ID Code.

Each frame must contain either a valid ID Code or a Full ID option or both. As long as the source of a frame does not know the ID Code expected by the destination (which is always the case for the first frame sent to a given member for a given group), the source must transmit the frame with the ID Code set to 0, and the Full ID option attached.

In what follows, we define the source member as the member that needs to learn the ID Code assigned by the destination member. Initially, the source member must be able to construct a complete Full ID option (that is, it must know all of the information about the destination member). The source member must transmit a Full ID option with each frame, with an ID Code of 0, until it learns the ID Code of the destination member through the reception of the Return ID Code transmitted in a frame by the destination member.

The destination member likewise must transmit frames with the Full ID option attached until it has determined that the source member has learned its ID Code. It knows this through the reception of a frame from the source member with the ID Code present.

The typical sequence of events is as follows. The source member initially has a frame to transmit to the destination member (the first frame transmitted between them for the given group). The source member assigns an ID Code, and transmits it in the Return ID Code field with the remainder of the fully composed Full ID option. The ID Code of this transmitted frame is set to 0.

The destination member, upon receiving the frame, records the ID Code of the source member for use in future transmitted frames. It assigns an ID Code of its own, and when it later has a return frame to send to the source member, it writes its assigned ID Code into the Return ID Code field of the Full ID option, and transmits the frame with the ID Code of the source member.

Upon receiving this frame, the source member records the ID Code of the destination member, and records also that the destination member has learned its ID Code. The next frame transmitted by the source member contains the ID Code of the destination member, and does not require a Full ID option. Upon receiving this frame, the destination also no longer needs to attach the Full ID option to its transmitted frames.

Thus, typically only two frames, one in each direction, require the Full ID option. This Full ID option is usually attached to frames initiated by a higher layer. If no such frames have been sent after repeated receptions of a Full ID from a neighbor member, a host may transmit an empty frame with the Full ID option attached.

The ID Code value assigned is a local matter. It must, however, be unique among all ID Codes currently assigned by the host. A newly assigned ID Code should not have been previously assigned and active for a very long time. This minimizes the likelihood that another host will still consider the previous assignment to be valid, and greatly simplifies management of codes.

The ID Code should have a significant random component to it, to make it hard to guess by third party hosts. A reasonable strategy would be to assign part of it as something useful for internally retrieving the appropriate data record (such as an index into an array), but for the rest of it to be a randomly generated number.

When a host receives a Full ID option with the Return ID Code set, it should store these fields, along with the ID information. Any frame sent in the return direction for the same ID information, except with the source and destination hosts reversed, may be sent using the received Return ID Code field in the ID Code field, and without the Full ID option attached.

A host should only retain a single ID Code for any given collection of ID information. If a received Full ID option indicates a new ID Code for ID information already stored, the old ID Code should be deleted.

A host should not blindly retain every new Return ID Code it receives. Ideally, the YIDP layer should get some indication from a higher layer that the ID Code should be retained. In particular, if the upper layer determines that a received frame is in error or otherwise should be ignored, then the YIDP layer should likewise ignore the ID Code. This obviously includes security checks. Further, if the upper layer determines that no relationship will be established between the two hosts, then the ID Code should also be ignored. It the case of YTMP, for instance, a valid relationship would be one between either neighbors of two members sharing pairwise knowledge.

The reason for this is to defend against a denial-of-service attack whereby the attacker sends a large number of different bogus Return ID Codes, thus causing valid and active ones to be erased and replaced by the bogus ones.

If a host receives a frame without a Full ID option attached, and the received ID Code is unknown, a frame with the corresponding ID Code Error Option attached should be returned. The host should remember when it has recently sent identical ID Code Error Options, and avoid sending more than a few in a given relatively short time period.

Upon receiving an ID Code Error Option, the corresponding locally stored ID Code information should be erased. Subsequent frames will contain a Full ID option, which will in turn trigger a Full ID option in the return direction with the correct ID Code.

A host may unilaterally erase any stored ID Code information at any time. Typically it won't do this unless the relationship with another host has clearly been ended (for instance, the host was a neighbor but quit), or unless no frames have been exchanged in either direction for a very long time. Prematurely erasing stored ID Code information is not a serious problem in any event. For transmitted frames, it simply means that a Full ID option has to be transmitted. For received frames, it means that an ID Code Error is returned, triggering a subsequent Full ID.

5.2 Assignment of the ID Code for IP Multicast Frames

Whereas assignment of the ID Code for unicast IP is an entirely local matter for the receiving host, for multicast IP it must be agreed upon by all members of the multicast group. This complicates its assignment.

On the other hand, it is here assumed that, when multicast IP is in use, it is because a cluster is using it. This means that the multicast IP group is local, and is not subject to NAT boxes, so the addresses seen by cluster members are stable. This simplifies the ID Code assignment compared to the unicast case.

Assignment of the actual value chosen for the ID Code is almost completely deterministic, as can be seen by the structure of the ID Code described in Section 4.

When a host first joins the cluster for a given group, it will know all of the ID Code except the 5-bit Resolver field. Until the ID Code is resolved (that is, contains a non-zero Resolver field) the host must transmit frames to the cluster with the Full ID option attached, and with the Resolver field set to value 0 (but with the rest of the ID Code properly constructed for the given group).

The cluster head is responsible for resolving the ID Code (assigning a non-zero value to the Resolver field). Generally, it will use the same value assigned by the previous cluster head. If it is the first cluster head, then it must resolve the ID code itself.

Two clusters are said to collide when they have the same IP group address and UDP port number, but belong to different groups. By the time a host becomes a cluster head, if there are colliding clusters, it will, with high probability, have heard from hosts from these clusters, and it will therefore know their Resolver values. The Resolver value selected by the host must be different from any of those heard.

A cluster head may select a Resolver value when it has transmitted 10 frames with unresolved ID Codes, over at least a 2 minute period, without having received any frames from colliding clusters that

1. also have unresolved ID Codes, and

2. have alpha-numerically higher Group IDs.

Until this happens, the cluster head may not select a Resolver value, and must continue using frames with unresolved ID Codes and Full ID options.

If a host with a resolved ID Code receives a frame from a host in a colliding cluster with an unresolved ID Code, the host sets a timer to a random value between 0 and 5 seconds. When the timer expires, it transmits an empty frame with its ID Code and Full ID option. If before the timer expires it receives any frame with its ID Code and Full ID option, then it cancels its timer.

If a host with an unresolved ID Code receives a frame from a host in a colliding cluster with an unresolved ID Code and an alpha-numerically lower Group ID, it likewise sets a timer, transmits an empty frame when the timer expires, and cancels the timer if it hears another such frame.

If a cluster head with a resolved ID Code receives a frame from a host in a colliding cluster with an identical (resolved) ID Code an an alpha-numerically higher group ID, then it must change its Resolver value. It does this by setting the Resolver value to 0, and proceeding as it normally does when deciding upon a Resolver value.

Finally, all cluster feet must use the ID Code of the latest frame received from the cluster head.

5.3 Port Numbers, IP Addresses, and Security

The TCP/UDP port number to be used for transmitting packets to another host may be derived one of two ways. First, it can be supplied by the calling application. Second, it can be derived from the TCP/UDP Source Port Number of the packet most recently received from that host.

Likewise, the IP address to be used for transmitting packets to another host may be derived one of two ways. First, it can be supplied by the application, or determined from a DNS lookup on the name supplied by the calling application. Second, it can be derived from the Source IP Address of the packet most recently received from that host.

In the case where the port number and IP address are derived from a received packet, the port number and IP address may or may not be that of the host that transmitted the packet. Rather, they may have been modified by a NAT box in transit. It is assumed, however, that they may validly be used to return packets to the host.

Strictly speaking, assuming that the information in the YIDP header can be trusted, the Source IP Address in a received packet is not necessary for identifying the sender of the packet. The Source Name in the IAP header supplies the necessary information. This means that the Source IP Address associated with a given ID Code may change. Whether or not this invalidates the ID Code depends on the security policy of the host and application.