

TCG Trusted Network Connect TNC IF-M: TLV Binding

Specification Version 1.0
Revision 30
1 February 2008
Draft

Contact:

Paul Sangster – Paul_Sangster@symantec.com (Editor, TNC Co-Chair)

Steve Hanna - shanna@juniper.net (TNC Co-Chair)

Work In Progress

This document is an intermediate draft for comment only and is subject to change without notice. Readers should not design products based on this document.

TCG

TCG Confidential

Copyright © TCG 2005-2008

Copyright © 2005-2008 Trusted Computing Group, Incorporated.

Disclaimers, Notices, and License Terms

THIS SPECIFICATION IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE. Without limitation, TCG disclaims all liability, including liability for infringement of any proprietary rights, relating to use of information in this specification and to the implementation of this specification, and TCG disclaims all liability for cost of procurement of substitute goods or services, lost profits, loss of use, loss of data or any incidental, consequential, direct, indirect, or special damages, whether under contract, tort, warranty or otherwise, arising in any way out of use or reliance upon this specification or any information herein.

This document is copyrighted by Trusted Computing Group (TCG), and no license, express or implied, is granted herein other than as follows: You may not copy or reproduce the document or distribute it to others without written permission from TCG, except that you may freely do so for the purposes of (a) examining or implementing TCG specifications or (b) developing, testing, or promoting information technology standards and best practices, so long as you distribute the document with these disclaimers, notices, and license terms.

Contact the Trusted Computing Group at www.trustedcomputinggroup.org for information on specification licensing through membership agreements.

Any marks and brands contained herein are the property of their respective owners.

Revision History

08/08/2007	First partial draft focused on use cases released.
08/15/2007	First full draft of use cases
08/22/2007	Initial release to TNC WG for review at Seattle F2F
10/7/2007	Updated based on IF-M design team discussion and added initial attribute proposals.
10/26/2007	80% complete version (except security protocol) for TNC virtual F2F discussion (r13)
11/5/2007	Updated 80% complete version addressing comments from vF2F (r14)
11/9/2007	Initial 100% complete draft except security functionality (v15)
11/19/2007	100% complete draft for TNC review (v18)
11/29/2007	Updated to include TNC review comments and to remove security parameters support (v20)
11/30/2007	Updated to address Lauren's comments (v21)
11/31/2007	Changed CMS Protected Attribute name and removed certificate chain
01/08/2008	Added couple attributes used by IF-M Security
01/10/2008	Modified attributes based on comments from Steve
1/15/2008	Final clean review for TNC e-ballot approval
1/31/2008	Updated to address comments from TC

Open Issues

This section tracks open issues for the duration of specification creation. This section will be removed prior to final draft.

#	Description	Comments
---	-------------	----------

Closed Issues

This section will be removed prior to final draft.

#	Description	Comments
PS-001	IF-IMC and IF-IMV use SMI number of 0 for TCG but TCG has an SMI number (21911). We should consider migrating to use our number instead of the reserved SMI 0.	PS – this might be used to recognize TNCCS attributes using the new namespace/format. Not sure if there exist any IF-M attributes to avoid in the TCG namespace but if so this might help here as well. Status - decided to leave as 0 which might be more attractive to the IETF NEA WG. Of course we could change to the TCG SMI until we know that the IETF doesn't plan to change the attributes used with SMI of 0.
SH-001	Should we require IF-M to have backward compatibility with existing IF-TNCCS implementations which assume a TLV start to the contents (instead of an IF-M header).	PS: The Message ID and Message Length used by IF-TNCCS should be maintained as 32 bits so as to not impact IF-TNCCS implementations. Inside the IF-M message (after the Message Length) we need to decide whether another TLV is required or whether we could just have a IF-M header and set of TLV attributes. If we have a fixed IF-M header it seems wasteful to have another TLV type

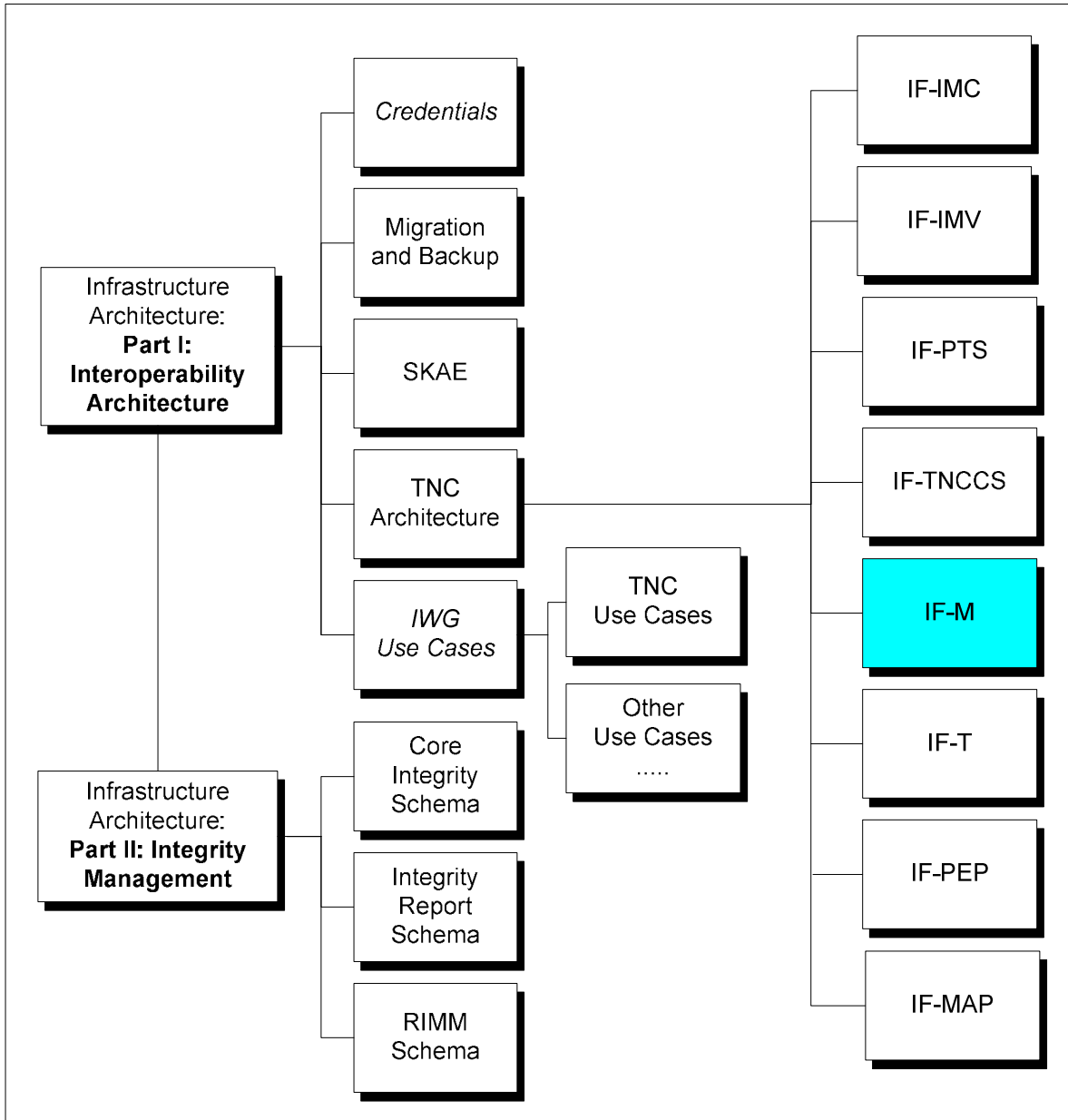
		<p>saying IF-M header and its length since it would be mandated by the protocol. This will come out in design discussions (not sure it impacts backwards compatibility with IF-IMC).</p> <p>SH-I definitely think we should maintain backward compatibility. Otherwise, IMCs and IMVs will not be able to use IF-M with existing TNCCs and TNCSs. In fact, an IMC will need to know not only that it's loaded on a TNCC that supports IF-M but also that it's talking to a TNCS that supports IF-M. I don't think that marking each IF-M message with a type will look odd, especially since I expect that IF-TNCCS 2.0 will use types to route messages to IMCs and IMVs (for compatibility and because it works well!).</p> <p>Status – don't believe we have a backward compatibility issue with IF-TNCCS 1.0 assuming it doesn't look inside the IF-M message but merely bases its routing on the Message Type (outside of IF-M).</p>
<p>SH-002</p>	<p>Steve proposes a simplification for IF-M where we state that only a single attribute is allowed to be in an IF-M message. This would mean that the IF-TNCCS message type could indicate exactly what attribute is contained in the message to improve message routing. This would enable only IMC/IMVs that know the attribute to receive it.</p> <p>If we opt for a single attribute per IF-M message then the attribute does not need to be encapsulated inside a TLV (type and length are indicated inside the IF-TNCCS protocol information).</p>	<p>PS – I'm not yet comfortable with limiting the IF-M payload to a single attribute. This would mean that IMC measurement reports where multiple items are returned would need to be sent as multiple IF-TNCCS message (each with their own type) or a single attribute with a complete structure defined.</p> <p>I had assumed we would have more granular attributes (like RADIUS) and we could group them in IF-M messages. This way an IMC only wishing to share some information could only send what it wishes to share rather than a structure with only some elements filled in (see use case 1-c and 3-b).</p> <p>If we drop use of TLV inside an IF-M message then we should change the name of this spec and it seems to blur the lines between the 2 protocols as IF-M is dependent on IF-TNCCS having the type and length protocol elements so we would need to state this as well as a requirement. Having it use its own TLVs, its more like SoH and is properly layered above IF-TNCCS without this dependency.</p> <p>Status – decided to leave IF-M as a TLV protocol inside of IF-TNCCS. IF-TNCCS will continue to expose the message type so it can route messages to the approp. IMC/IMVs. Inside each IF-TNCCS message can be a list of attributes but they must all be associated with the component described by the message</p>

		type (so proper message routing occurs).
PS-002	Current protocol has the granularity of assessment information constrained to the component level. For some components this is probably fine, but the 'operating system' component may be too broad. Assessment might wish to evaluate sub-components such as a web server version.	Status – we decided to limit the initial set of components and let IWG take the lead on operating system sub-component evaluation. The NEA WG might also wish to expand the list but since TNC is on a tight schedule let keep the list small for now.
PS-003	Is a 8 bit component (sub)type and an 8 bit attribute (sub)type enough to enumerate all the components and attributes. Even though we support an SMI for vendor and standard name space, IWG and I believe these are too small for the standards space (over the long term).	Status – we decided at the vF2F that we would make all the sub-type fields in TNCCS and IF-M use 32 bit SMIs and 32 bit sub-type fields to avoid any name space exhaustion issues. These are much cleaner when we go to NEA as well. Note that this means our Message IDs are now 8 octets (larger than the 32 bits documented in the past). We will include optional API extensions to allow for IMCs/IMVs to discover the longer values.
SH-004	Question whether the TCG should host a registry for certain key vendor-defined values such as OS Family so it's easier for 3 rd party IMC/IMV developers to know what to expect.	Status – group discussed this during vF2F and decided to wait on requesting the registry until we had enough critical mass to make it valuable. 3 rd party developers can learn other vendors values via plugfests and interop testing.
SH-006	If a message contained multiple Product Information and Version attribute how would the recipient correlate which Version go with which Product Information. We either need to move Version information into Product Information or figure out a more general approach.	Status – discussed this at vF2F and suggested we include a correlation ID that will uniquely (per IMC) identify attributes (generally) when multiple of the same attribute type are returned in a message. This allows the IMV to match up attributes involving the same product without grouping the attributes in the same attribute. (See attribute header's Correlation ID).
SH-005	Current spec includes support for an EXCL flag that all IMC and IMV must check before processing each message to learn whether the IF-M message is for it. In order for this to work all IMC/IMV MUST do this or else an entity might operate on a message that was only supposed to be processed by another entity. For example remediation instructions going to 2 firewall IMCs where the IMV only wanted one to perform the remediation. This feature could be offered by an enhanced IF-TNCCS protocol so that only the targeted IMC or IMV would receive the message. This enables the sender to define a destination for the message that would bypass the other subscribed parties. This is a new concept for TNC so would require text changes in several	Status – moving this features to IF-TNCCS

	specs and impact the TNCCS protocol and IF-IMC and IF-IMV APIs. Do we want to move this feature to TNCCS so we no longer have to require all IMC/IMV to check this flag (perhaps improving overall security and reliability).	
PS-004	Naming question about what to call the existing "Message Type" used by TNCC and TNCS to route messages. As we worked on IF-M it became clear that this is really more of a component type indication. Should we change the name in the new specs (IF-M 1.0 and IF-TNCCS 2.0) to call this a component type when housed inside a IF-M TLV? I believe this naming will make more sense to NEA. I'm also open to other names like "Software Type" or "Subsystem ID" ...	<p>This change would cause a naming inconsistency with 1.x IF-TNCCS. I've looked at the architecture spec and it only uses Message Type informally (and not consistently). We already have to change this text since we're moving to 64 bit message types so this change isn't a problem. IF-IMC and IF-IMV could continue to call it Message Type which is more generic but these already have to describe the new 64 bit values (which could be called component types) since the APIs don't have a way to pass such a value yet.</p> <p>IF-TNCCS 2.0 would be easy to change just leaving IF-TNCCS 1.x with the old naming. This would allow us to talk about each in the architecture document and explain their slightly different semantics and lengths.</p> <p>Status – We agreed to call the subtype field of the message type the component type in this specification when we're talking about IF-M standard messages. Vendor-defined messages can use some other semantic for this field.</p>
SH-003	Based on feedback from Steve changed the remote IMV use case in section 2.2.3 to focus on a remote TNCS and IMV away from the NAA where IF-T is terminated. Steve suggests this is a more likely use case (e.g. eduroam).	<p>PS – Defining the use case as having a remote TNCS with a bridged IF-T session doesn't highlight the need for IF-M security. In fact with IF-T being bridged the security model is very similar to the standard model since the proxying TNCS needs to proxy the entire IF-T handshake including any certificate exchanges. Even if the TNCS is creating a 2nd IF-T session and performing proxying it doesn't highlight the need for IF-M security.</p> <p>I don't feel very strongly about the remote IMV use case but it does best show the use for end to end IF-M security. I realize there may be other ways to protect the TNCS to IMV hop so will also highlight the use of end to end authentication as well.</p> <p>Status – text in 2.2.3 changed to reflect other use cases for IF-M security.</p>
SH-007	Section 3.4 describes the general message flows for an IF-M dialog. These flow examples highlight a couple of fairly complex usages that might require the reader to already understand the semantics of the correlation ID and other	<p>PS – Another option is to make the description a little higher level. My preference would be to not move it as section 3.4 is trying to establish some general context for IF-M before going into the protocol details. However I do agree</p>

	<p>protocol fields. Maybe this section should be moved later in the spec (after the protocol definition) so the reader is aware of these concepts.</p>	<p>that example 2 is more difficult to understand without understanding why this is a protocol issue.</p> <p>Status – moved to the end of the spec for consistency with other TNC specs.</p>
--	--	--

IWG TNC Document Roadmap



Acknowledgement

The TCG wishes to thank all those who contributed to this specification. This document builds on considerable work done in the various working groups in the TCG.

Special thanks to the members of the TNC contributing to this document:

Scott Kelly	Aruba Networks
Mahalingam Mani	Avaya
Hidenobu Ito	Fujitsu Limited
Sung Lee	Fujitsu Limited
Kazuaki Nimura	Fujitsu Limited
Mauricio Sanchez	Hewlett-Packard
Han Yin	Huawei
Diana Arroyo	IBM
Stuart Bailey	Infoblox
Ravi Sahita	Intel Corporation
Ned Smith	Intel Corporation
Steve Hanna (TNC co-chair)	Juniper Networks, Inc.
John Jerrim	Lancope, Inc.
Ryan Hurst	Microsoft
Sandilya Garimella	Motorola
Meenakshi Kaushik	Nortel Networks
Paul Sangster (Editor, TNC co-chair)	Symantec
Greg Kazmierczak	Wave Systems
Thomas Hardjono	Wave Systems

Table of Contents

1	Scope and Audience	12
2	Background	13
2.1	Purpose of IF-M	13
2.2	Supported Use Cases	13
2.2.1	TNCC Initiated Assessment Use Case	13
2.2.2	TNCS Initiated Assessment Use Case	14
2.2.3	IF-M Involving Remote IMV Use Case	15
2.3	Non-supported Use Cases	16
2.4	Requirements	16
2.5	Non-Requirements	17
2.6	Assumptions	18
2.7	Keywords	18
2.8	IF-M Message Diagram Conventions	18
3	IF-M Message Protocol	19
3.1	IF-M Messaging Model	19
3.2	IF-M Relationship to IF-TNCCS	19
3.3	IF-M Messages in IF-TNCCS	20
3.4	IF-M Component Types	21
3.5	IF-M Message Header Format	21
4	IF-M Attributes	23
4.1	IF-M Attribute Header	23
4.2	TNC Standard Attributes	25
4.2.1	Attribute Applicability	27
4.2.2	Attribute Request	28
4.2.3	Product Information	29
4.2.4	Numeric Version	30
4.2.5	String Version	31
4.2.6	Operational Status	32
4.2.7	Port Filter	33
4.2.8	Installed Packages	34
4.2.9	IMV Assessment Results	35
4.2.10	Remediation Instructions	36
4.2.11	IF-M Error	38
4.3	Vendor-Defined Attributes	41
5	Security Considerations	42
5.1	Trust Relationships	42
5.1.1	IMC	42
5.1.2	IMV	42
5.1.3	TNCC, TNCS and IF-TNCCS	42
5.2	Security Threats	43
5.2.1	Attribute Theft	43
5.2.2	Message Fabrication	43
5.2.3	Attribute Modification	43
5.2.4	Attribute Replay	44
5.2.5	Attribute Insertion	44
5.2.6	Denial of Service	44
6	Privacy Considerations	45
7	IF-M Message Flows	46
7.1	Simple IMC Initiated Example	46
7.2	IMV Initiated with Complex IMC Example	47
7.3	Multiple IMCs for Single Component Example	47
8	References	49
8.1	Normative References	49

8.2 Informative References49

1 Scope and Audience

The Trusted Network Connect Work Group (TNC-WG) has defined an open solution architecture that enables network operators to enforce policies regarding the security state of endpoints in order to determine whether to grant access to a requested network infrastructure. This security assessment of each endpoint is performed using a set of asserted integrity measurements covering aspects of the operational environment of the endpoint. Part of the TNC architecture is IF-M, a standard protocol between Integrity Measurement Collectors on the TNC Client to the Integrity Measurement Verifiers on the TNC Server. This document defines and specifies standard messages for the IF-M protocol.

Architects, designers, developers and technologists who wish to implement, use, or understand IF-M should read this document carefully. Before reading this document any further, the reader should review and understand the TNC architecture as described in [IF-ARCH].

2 Background

2.1 Purpose of IF-M

This document describes and specifies IF-M, an application level protocol capable of carrying Integrity Check Handshake messages between the Integrity Measurement Collectors (IMCs) on the TNC Client and the Integrity Measurement Verifiers (IMVs) on the TNC Server. IF-M is a multiple roundtrip messaging protocol that enables IMC(s) to send measurement information about local components on the endpoint to IMV(s) for evaluation against network security policy. The IMV(s) can respond using IF-M messages with additional requests for measurement data or optionally with its IMV Action Recommendation. The decision might also include a set of remediation instructions that the IMC could perform to bring its associated component into compliance with the IMV's policy.

The IF-M protocol is carried over the network by the IF-TNCCS protocol [IF-TNCCS][IF-TNCCS-SOH]. The TNC Client and Server pass the received IF-M messages to the appropriate set of IMCs and IMVs using the respective IF-IMC[IF-IMC] and IF-IMV[IF-IMV] APIs.

This specification defines the standard IF-M messages that can be used to enable interoperability between an IMC from one vendor and an IMV from another vendor. However, these messages are only a subset of the broader IF-M protocol. For years prior to the IF-M specification, IMC and IMV vendors have been employing vendor-specific IF-M messages to communicate between their IMCs and IMVs. It is expected that in the future a mixture of vendor-specific IF-M messages and standard IF-M messages may be employed to provide the best of both worlds: interoperability between IMCs and IMVs from different vendors and the tight integration that vendor-specific IF-M messages provide.

IF-M is envisioned to be a robust, extensible protocol capable of exchanging sets of attributes between zero or more subscribed IMCs and IMVs. This specification includes the definition of the message protocol and a partial set of standard attributes that are expected to have general applicability to different types of components (e.g. version information). IF-M attribute name space is extensible allowing for vendor-defined attributes to be established that convey product specific information and to allow for additional standard attributes to be defined in future specifications as more deployment experience becomes available.

2.2 Supported Use Cases

This section describes the IF-M use cases that must be supported. In order to focus on the protocol interactions, these use cases do not describe what triggered the assessment to occur. Thus it's expected that each use case is able to work regardless of what triggers the TNCC or TNCS to start the assessment. For example, such triggers include: an endpoint initially joining the network, a change to an endpoint or a significant event on the endpoint, the TNCS's policy change or the TNCS receiving notice of a significant event. Each type of trigger must be able to cause an assessment to occur even if a prior assessment of the endpoint has already occurred. In each case, the following use case message exchanges must be supported.

2.2.1 TNCC Initiated Assessment Use Case

- 1) TNCC determines that an assessment of the endpoint is required. This might occur if the TNCC becomes aware that the endpoint is trying to access an 802.1X protected network.
- 2) TNCC invokes one or more IMCs to send measurement information to their corresponding IMVs on the TNCS. Each IMC consults its policy and could choose to:
 - a) Send nothing, if unwilling or unable to participate
 - b) Send a hello message indicating its availability to respond to requests
 - c) Send a subset of the measurements it's able to collect
 - d) Send all of its collected measurements

- e) Send previously received and cached assertion attributes
- 3) TNCS passes attributes sent by IMCs to IMVs that have expressed an interest in the measurement information received from the TNCC. Each IMV consults its assessment policy and could choose to:
- a) Send nothing (e.g. might just be in monitoring or audit mode)
 - b) Request additional measurement information from the IMC(s) by sending one or more IF-M messages. These IF-M messages could provide session specific information that should be bound into the measurement response (e.g. an IMV providing a nonce to be used during a TPM_Quote by the IMC). This is the expected behavior for each received hello message from an IMC. In response to the IMV requests for measurement information, the IMCs repeat step 1 above. This could lead to additional IMV requests in step 2.
 - c) Make an assessment decision based on the measurements provided, return the IMV Evaluation Result to the TNCS and take one or more of the following actions:
 - i) Send no IF-M messages
 - ii) Send the component level IMV Action Recommendation to the IMCs
 - iii) Send remediation instructions to the IMCs
 - iv) Send assertion attributes to the IMCs. The assertion attributes could describe the level of compliance determined by the IMV.

2.2.2 TNCS Initiated Assessment Use Case

- 1) TNCS determines that an assessment of an endpoint is required. For instance, this could occur due to the TNCS becoming aware of an event occurring, due to a policy requiring periodic reassessment, or due to a TNCS policy change.
- 2) TNCS invokes one or more IMVs to initiate the assessment. Each invoked IMV could send requests for measurement information to their corresponding IMCs on the TNCC. Each invoked IMV consults its policy and could choose to:
- a) Send no IF-M messages, if unwilling or unable to participate
 - b) Send a request for measurement information attributes. This information request could also include sending session specific information that should be bound into the measurement response (e.g. IMV providing a nonce to be used during a TPM_Quote by the IMC).
- 3) TNCC passes the attribute requests sent by IMVs to IMCs that have expressed a willingness to provide the requested measurement information. Each IMC consults its attribute policy and could choose to:
- a) Send no IF-M messages, if unwilling or unable to participate
 - b) Send a subset of the requested measurement attributes that it's able to collect (possibly factoring in privacy policy)
 - c) Send all of the requested measurement attributes
 - d) Send previously received and cached assertion attributes (possibly in addition to some requested measurement attributes)
- 4) TNCS passes attributes sent by IMCs to IMVs that have expressed an interest in the measurement information received from the TNCC. Each IMV consults its assessment policy and could choose to:
- a) Send no IF-M messages (e.g. might just be in monitoring or audit mode)
 - b) Request additional measurement information from the IMC(s) by sending one or more IF-M messages. For example, this might occur if the initial roundtrip was to determine the platform type and operating system information and this next message will request specifics potentially unique to this endpoint. In response to the IMV requests for measurement information, the IMCs repeat step 3 above. This could lead to additional IMV requests in step 4.

- c) Make an assessment decision based on the measurements provided and then return the IMV Evaluation Result to the TNCS and take one or more of the following actions:
 - i) Send no IF-M message
 - ii) Send the component level IMV Action Recommendation to the IMCs
 - iii) Send remediation instructions to the IMCs
 - iv) Send assertion attributes to the IMCs. The assertion attributes could describe the level of compliance determined by the IMV.

2.2.3 IF-M Involving Remote IMV Use Case

This use case augments the TNCC and TNCS Initiated Assessment use cases above by moving the IMVs from being local to the TNCS and NAA to existing remotely on another system on the network. This use case highlights the potential security exposure of having IMVs remote to the NAA and possibly the TNCS thus potentially requiring message protection in order to make it equivalent to when the entire TNC server is on a single system. Because these messages potentially contain security sensitive information (e.g. remediation instructions or patch state of the endpoint) they may require equivalent protection to when the TNC Server components are co-located on a single system.

In order to not repeat all of the other use cases with just a single alteration, this section bases its description on section 2.2.1 and highlights the significant differences using underlining. In particular, this use case removes the assumption that the TNC server components are all co-located on a single system (e.g. IMVs are remote from the TNCS). The following description assumes that at least one IMV is located remotely from the TNCS and no other security is provided on the link between the TNCS and IMV. An alternative solution to the remote IMV security concern is to use a secure protocol between the TNCS and the remote IMV. However, IF-M security may also be useful if the IMC or IMV does not trust the TNCC and TNCS to see unencrypted messages or if the IMV wishes only to accept information from a recognized authenticated IMC on the endpoint that is known to be reliable for both reporting measurements and performing remediation. This might not be the case for all IMC registered to receive message about a component on some endpoints.

- 1) TNCC invokes one or more IMCs to send measurement information to their corresponding IMVs on the TNCS. Based on the target network, the IMCs might have a policy indicating that security protections are required since the network's IMV require IMC authentication, integrity and confidentiality protection when crossing the unprotected network to the IMV.. Each IMC consults its policy and could choose to:
 - a) Send nothing, if unwilling or unable to participate
 - b) Send a hello message indicating its availability to respond to requests
 - c) Send a subset of the measurements it is able to collect. Because these messages are transported over an untrustworthy network between the TNCS and the IMV, they might require end-to-end security protection.
 - d) Send all of its collected measurements. Because these messages are transported over an untrustworthy network between the TNCS and the IMV, they might require end-to-end security protection.
 - e) Send previously received and cached assertion attributes possibly protected by end-to-end security protection.
- 2) TNCS passes attributes sent by IMCs to the remote IMVs that have expressed an interest in the measurement information received from the TNCC. The IMV exists remotely from the TNCS where the IF-T protections have been terminated so the IF-M messages need protection. The remote IMV may require authentication of the sending IMC to decide whether the IMC's information is considered reliable particularly when multiple IMCs are reporting the same attributes but with different values. Each IMV consults its assessment policy and could choose to:
 - a) Send no IF-M messages

- b) Request additional measurement information from the IMC(s) by sending one or more IF-M messages. In response to these IMV requests for measurement information, the IMCs repeat step 1 above. This could lead to additional IMV requests in step 2.
- c) Make an assessment decision based on the measurements provided, return the IMV Evaluation Result to the TNCS and take one or more of the following actions:
 - i) Send no IF-M messages
 - ii) Send the component level IMV Action Recommendation to the IMCs. This exposes the IMV Action Recommendation to attack on the network between the TNCS and IMV so security protection may be required.
 - iii) Send remediation instructions to the IMCs – this exposes the remediation instructions to attack on the network between the TNCS and IMV so security protection may be required. Also the IMV may wish to establish a private, authenticated session with a particular IMC to assure that the proper IMC performs the remediation (when multiple IMC for a component exist on an endpoint). In this case the IMC also benefits by being able to authenticate the IMV in case multiple IMVs are providing remediation instructions.
 - iv) Send assertion attributes to the IMCs. The assertion attributes describe the level of compliance determined by the IMV. This exposes the assertion attributes to various attacks on the unprotected network between the TNCS and IMV so may require security protection to be employed.

Therefore in order to be able to safely send over the network the IF-M messages, possibly including endpoint measurements, remediation instructions, and even assertion attributes, the IF-M protocol needs to provide additional security protections to safeguard the information as well as if the IMV were local to the TNCS and NAA.

2.3 Non-supported Use Cases

- None

2.4 Requirements

Here are the requirements that IF-M must meet in order to successfully play its role in the TNC architecture.

- Flexibility

The IF-M protocol MUST support all the functions and use cases described in the TNC architecture as they apply to the communications between the IMC and IMV. The IF-M protocol MUST allow either the IMC or IMV to initiate the assessment or reassessment when operating over a usable IF-TNCCS session. When the IMC initiates the assessment, the IMV MUST allow the IMC to proactively send measurements prior to the IMV sending a measurement request.

The IF-M protocol MUST be capable of supporting multiple round trip message exchanges during an assessment or reassessment. This allows the IMVs to send multiple requests for measurements potentially based on the results of earlier requests (e.g. based on the endpoint's operating system).

IF-M attributes MUST be capable of containing a wide variety of types of data values including: binary data, encrypted or compressed data, and textual strings. Any string included in IF-M intended for user display MUST be able to be encoded in the user's preferred language (when known). IF-M MUST be able to carry standard defined attributes and/or vendor-defined attributes.

- Secure

IF-M protocol MUST provide the capability to protect its messages end to end between the IMC and IMV. This protection MUST guard against active and passive attackers by offering bi-directional authentication, detection of alteration or replay of the messages, and confidentiality of the message

contents as mandated by deployment policy. IF-M security protections enable IMVs existing on a system remote from the termination of the IF-T connection at the NAA to have end to end protected communications with IMCs. This could be particularly important when security sensitive information such as remediation instructions are sent in IF-M messages. See the Security Considerations section of this document for more details on security requirements.

- Efficient

The TNC architecture delays network access until the endpoint is determined to not pose a security threat to the network based on its asserted integrity information. To minimize user frustration, the IF-M protocol MUST minimize delays and make IF-M communications as rapid and efficient as possible. Efficiency is also important when you consider that some network endpoints are small and low powered, some networks are low bandwidth and/or high latency, and some IF-T protocols only allow one packet in flight.

- Transport Independence

IF-M protocol MUST be agnostic of the underlying IF-T transport protocol and thus not change in message format when different IF-T protocols are used. However, IMCs and IMVs may alter their level of verbosity (payload size) in the IF-M message when faced with underlying protocols which are bandwidth constrained.

- Extensible

IF-M protocol and the attributes contained within it MUST be very extensible allowing for additional protocol capabilities and large numbers of attributes to be defined by future specifications. The attribute name space MUST support large numbers (hundreds) of vendor specific attributes for each vendor without collisions and large numbers (hundreds) of standard defined attributes which can be defined over time.

- Scalable

IF-M protocol MUST be capable of housing a large number (hundreds) of attributes in a single message exchange and allow for use of attributes with large attribute values (tens of kilobytes). This capability might not be practical or even necessary for all deployments (e.g. low bandwidth, high latency, time sensitive environments) but should be possible without alteration of the base protocol. IMCs and IMVs may choose to scale back the number and size of the attributes sent based on other factors (IF-T considerations, privacy filters, etc.).

- Backward Compatibility

IF-M protocol SHOULD be backwards compatible with the existing IF-IMC and IF-IMV APIs and TNCC and TNCS. This requirement primarily covers the outer IF-M message envelopes which are used by the TNCC and TNCS to route messages and by the IF-IMC and IF-IMV APIs which pass the values to the subscribed components.

2.5 Non-Requirements

Here are certain requirements that IF-M explicitly is not required to meet. This list is not exhaustive (complete).

- Compression

IF-M protocol will not automatically compress large messages to improve their suitability for particularly network limitations (e.g. bandwidth, latency). This is the responsibility of the IMC and IMV if it is to be provided.

2.6 Assumptions

Here are the assumptions that the IF-M protocol makes about other components in the TNC architecture.

- Reliable Message Delivery

The TNC Client and TNC Server are assumed to provide reliable delivery for IF-M messages sent between the IMCs and the IMVs. In the event that reliable delivery cannot be provided, the TNC Client or TNC Server is expected to terminate the connection.

2.7 Keywords

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in RFC 2119 [KEYWORDS]. This specification does not distinguish blocks of informative comments and normative requirements. Therefore, for the sake of clarity, note that lower case instances of must, should, etc. do not indicate normative requirements.

2.8 IF-M Message Diagram Conventions

This specification defines the syntax of the IF-M message header and the standard set of attributes using diagrams. Each diagram depicts the format and size of each field in bits. Implementations MUST send the bits in each diagram as they are shown from left to right for each 32-bit quantity traversing the diagram from top to bottom. Multi-octet fields representing numeric values must be sent in network (big endian) byte order. Descriptions of bit fields (e.g. flags) values are described referring to the position of the bit within the field. These bit positions are numbered from the most significant bit through the least significant bit so a one octet field with only bit 0 set has the value 0x80.

3 IF-M Message Protocol

This section discusses the use of the IF-M message and its attributes within the TNC architecture and specifies the syntax and semantics for the IF-M message header. The details of each attribute included within the IF-M payload are specified in section 4.

3.1 IF-M Messaging Model

IF-M messages are carried by the IF-TNCCS protocol which provides a multi-roundtrip reliable transport and end to end message delivery to subscribed (interested) parties using a variety of underlying network protocols. IF-M is unaware of these underlying IF-T transport protocols being used below IF-TNCCS. The interested parties consist of IMCs on the TNCC and IMVs associated with the TNCS that have registered to receive messages about particular types of components (e.g. anti-virus) during an assessment. The IF-M messaging protocol operates synchronously within an assessment session with IMCs and IMVs taking turns sending one or more messages to each other. Each IF-M message may contain one or more attributes associated with the functional component defined in the IF-TNCCS protocol. IMCs may only send IF-M messages to IMVs and vice versa. No IMC to IMC or IMV to IMV messaging is allowed to occur. Each IMC or IMV may send several IF-M messages in succession before indicating that it has completed its response to the TNCC or TNCS respectively. As necessary, the TNCC and TNCS will batch these messages prior to sending over the network.

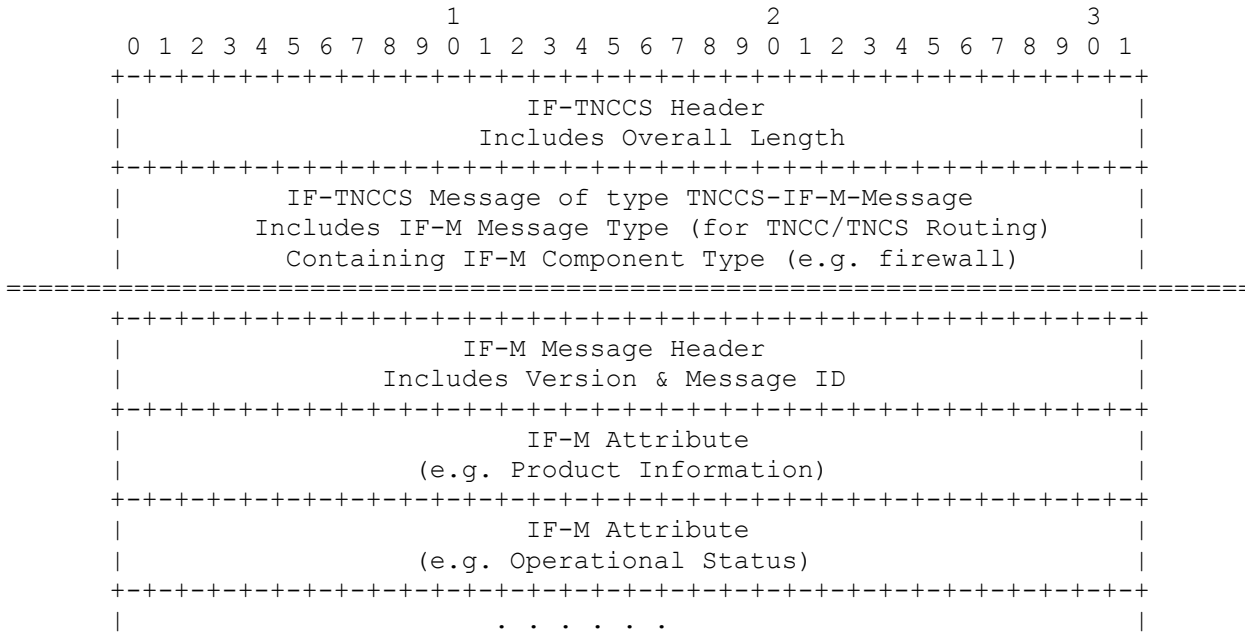
IF-TNCCS provides a message publish/subscribe model of message exchange. This means that, at any given point in time, zero or more subscribers for a particular type of message may be present on a TNCC or TNCS. This is beneficial since it allows one IMC or IMV to combine multiple functions (like anti-virus and personal firewall) by subscribing to both TNC standard component types and also allows multiple IMCs or IMVs to support the same components such as two anti-virus IMVs that are each used to manage their own respective anti-virus client software. However, this publish/subscribe model has some possible negative side effects. When an IMC or IMV initially sends an IF-M message, it does not know whether it will receive many, one or no IF-M messages from the other side. For many types of assessments, this is acceptable but in some cases a more direct channel binding between a particular IMC and IMV pair is necessary. For example, an IMV may wish to provide remediation instructions to a particular IMC that it knows is capable of remediating a non-compliant component. This can be accomplished using the IF-TNCCS capability to limit distribution of a message to a single IMC.

3.2 IF-M Relationship to IF-TNCCS

This section summarizes the major elements of an IF-M message as they might appear inside of an IF-TNCCS message. The double line (===) in the diagram below indicates the separation between the IF-TNCCS and IF-M protocols. The IF-M portion of the message is delivered to each IMC or IMV registered to receive messages containing a particular message type. Note that IF-TNCCS is capable of carrying multiple IF-TNCCS and IF-M messages in a single IF-TNCCS batch. See the IF-TNCCS specification for more information on its capabilities [IF-TNCCS].

One important linkage between the IF-M and IF-TNCCS protocols is the message type that is used by the TNCC and TNCS to route messages to interested IMCs and IMVs. The message type indicates the software component (component type) that is associated with the attributes included inside the IF-M message. Therefore, IMCs and IMVs written to support an assessment of a particular component can register to receive messages about the component and thus participate in its assessment. Each IMC and IMV MUST only send IF-M messages containing attributes that pertain to the software component defined in the message type of the message. This assures that only the appropriate IMCs and IMVs that support a particular type of component will receive attributes related to that component. If a message contained a mix of attributes about different components and a message type of only one of those components, the message would only be delivered to parties interested in the component type included in the message type, so other interested recipients wouldn't see those attributes.

The message type is comprised of 2 fields: a Vendor ID and a message subtype. The Vendor ID identifies the vendor or other organization that defined this message type. The message subtype identifies the message type more particularly within the set of message types defined by that vendor. This specification defines several standard message subtypes to be used with the TCG's Vendor ID. Within this specification, the subtype field is used to indicate the type of component (e.g. firewall) involved with the message's attributes. Therefore for clarity the message subtype field will be referred to as the "component type" in this specification. Vendor-defined name spaces may use other semantics for the subtype field as this is outside the scope of this specification.



Overview of an IF-TNCCS batch that contains an IF-M Message

For example, if a TNCC sent an IF-TNCCS batch that contained an IF-M message with a message type indicating firewall component, this message would be routed by the TNCS to IMVs registered to assess firewalls. Each registered IMV would receive a copy of the IF-M message including the IF-M header and set of attributes. It is important that each of the attributes included in the IF-M message be associated with the firewall component because only the firewall interested IMC and IMV will receive the message. For example, if the above message contained both firewall and operating system attributes (inside an IF-M message with a component type of firewall), then any IMC and IMV registered to receive operating system messages would not receive those attributes as the messages would only be delivered to those registered for firewall messages.

3.3 IF-M Messages in IF-TNCCS

As depicted in section 3.2, an IF-M message consists of an IF-M header followed by a sequence of one or more attributes. The IF-M message header (described in section 3.5) and the header for each of the IF-M attributes (specified in section 4.1) have a fixed type-length-value (TLV) format. Each IF-M message MAY contain a mixture of standards based and vendor defined attributes identifiable using the type portion of the attribute header. All IMCs and IMVs compliant with this specification MUST be capable of processing multiple attributes in a received IF-M message. IMCs or IMVs that receive an IF-M message can use the attribute header's length field to skip any attributes that it does not understand unless the attribute is marked as mandatory to process.

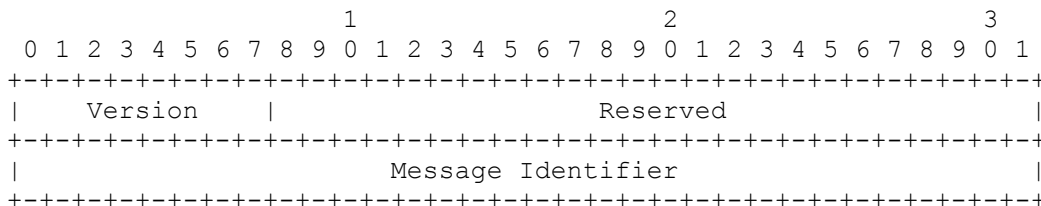
3.4 IF-M Component Types

This section defines the component type values used within the message subtype field for IF-M Messages that have the TCG SMI Private Enterprise Number (PEN). These TNC standard values are present in the message type field of the IF-TNCCS protocol (TNCCS-IF-M-Message) to describe which type of component is associated with the IF-M attributes included within the message. The message type field is used by the TNCC and TNCS to route IF-M messages to IMCs and IMVs that have registered to receive messages containing the message type. This allows for IMCs and IMVs to receive messages about specific types of component. More component types will likely be added in the future.

Component Name	TNC Standard Component Type	Description
Reserved	0x00000000	Reserved for use in specification examples, experimentation and testing.
Operating System	0x00000001	Operating system running on the endpoint
Anti-Virus	0x00000002	Host-based anti-virus software
Anti-Spyware	0x00000003	Host-based anti-spyware software
Anti-Malware	0x00000004	Host-based anti-malware (e.g. anti-bot) software not included within anti-virus or anti-spyware components
Firewall	0x00000005	Host-based firewall
Intrusion Detection/Prevention (IDPS)	0x00000006	Host-based intrusion detection and/or prevention software
Virtual Private Networking (VPN)	0x00000007	Host-based VPN software

3.5 IF-M Message Header Format

This section describes the format and semantics of the IF-M header. Every TNC standard IF-M message MUST start with an IF-M header. The IF-M header provides a common context applying to all of the attributes contained within the IF-M payload. The payload consists of a sequence of assessment attributes described in section 4.



Header Field	Description
Version	This field indicates the version of the format for the IF-M

	<p>message. This version is intended to allow for evolution of the IF-M message header and payload in a manner that can easily be detected by message recipients.</p> <p>IF-M message senders MUST set this field to 0x01 for all IF-M messages that comply with formats and requirements described in version 1.0 of this specification. Implementations responding to an IF-M message containing a supported version SHOULD use the same Version number to minimize the risk of version incompatibility.</p> <p>Message senders MAY send an empty IF-M message with the Version value set to 0x00 in order to discover the IF-M protocol versions supported by peer recipients (see IF-M Error message information in section 4.2.11). Message recipients MUST NOT interpret the contents (after the Version field) of an IF-M message containing a version number that the recipient does not support. Message recipients MUST return a TNC_IFM_VERSION_NOT_SUPPORTED error in an IF-M error message upon receipt of a message containing an unsupported version including 0x00 used for version discovery.</p> <p>IF-M message initiators supporting multiple IF-M protocol versions SHOULD be able to alter which version of IF-M message they send based on prior message exchanges with a particular peer IMC or IMV.</p>
Reserved	Reserved for future use. This field MUST be set to zero on transmission and ignored upon reception.
Message Identifier	<p>This field contains a value that uniquely identifies the message from a particular IF-M message sender. This value can be included in a response message to indicate which message was received and caused the response. For example, this field is included in the TNC error messages so the recipient can determine which message caused the error.</p> <p>IF-M message senders MUST NOT send the same message identifier during an assessment. Message identifiers may be randomly generated or sequenced as long as values are not repeated during an assessment message exchange.</p>

4 IF-M Attributes

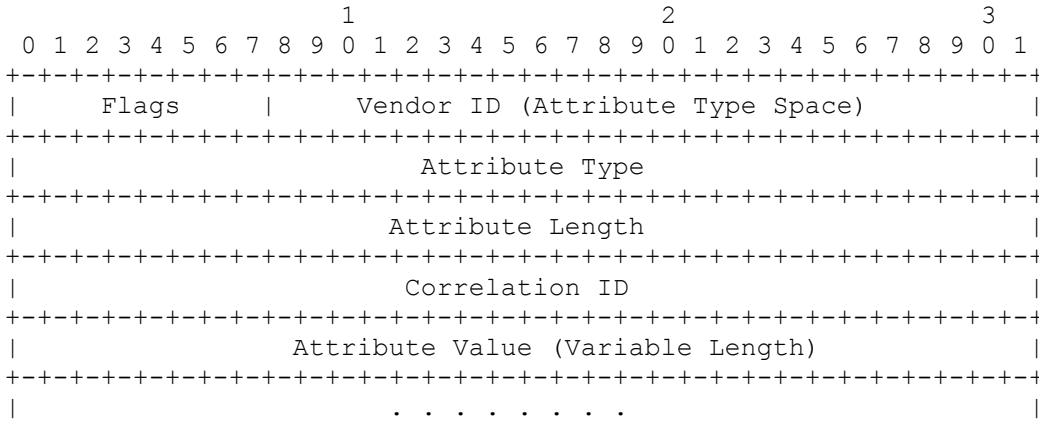
This section defines the IF-M attributes that can be carried within an IF-M message. The initial section defines the standard attribute header that appears at the start of each attribute in an IF-M message. The second section defines each of the TNC standard attributes and the final section discusses how vendor-defined attributes can be used within an IF-M message. Vendor-defined attributes are those defined outside of this specification and use the vendor's SMI Private Enterprise Number in the Attribute Type field.

An IF-M message MUST contain an IF-M header (defined in section 3.4) followed by a sequence of zero or more IF-M attributes. All IF-M attributes MUST begin with a standard IF-M attribute header, as defined in section 4.1. The contents of IF-M attributes vary widely, depending on their attribute type. Section 4.2 defines the TCG standard attributes. Section 4.3 discusses how vendor-specific attributes can be defined.

4.1 IF-M Attribute Header

Following the IF-M message header is a sequence of zero or more attributes. All IF-M attributes MUST begin with the standard IF-M attribute header defined in this subsection. Each attribute described in this specification is represented by a TLV tuple. The TLV tuple includes an attribute identifier comprised of the Vendor ID and Attribute Type (type), the TLV tuple's overall length and finally the attribute's value. The use of TLV representation was chosen due to its flexibility and extensibility and use in other standards. Recipients of an attribute can use the attribute type fields to determine the precise syntax and semantics of the attribute value field and the length to skip over an unrecognized attribute. The length field is also beneficial when a variable length attribute value is provided.

The TLV format does not contain an explicit TLV format version number, so every attribute included in a particular IF-M message MUST use the same TLV format. Using the IF-M message version number to indicate the format of all TLV attributes within an IF-M message allows for future versioning of the TLV format in a manner detectable by IF-M message recipients. Similarly, requiring all TLV attribute formats to be the same within an IF-M message also assures that recipients compliant with a particular IF-M message version can at least parse every attribute header and use the length to skip over unrecognized attributes. Every IF-M version 1.0 compliant TLV attribute MUST use the following TLV format:



TLV Field	Description
-----------	-------------

Flags	<p>This field defines flags impacting the processing of the associated attribute. The following table defines the bit encodings for each flag starting from the left to right:</p> <table border="1" data-bbox="402 317 1373 1409"> <thead> <tr> <th data-bbox="407 317 609 380">Bit Encoding</th> <th data-bbox="609 317 1369 380">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="407 380 609 1073"> Bit 0 No Skip Flag (NOSKIP) </td> <td data-bbox="609 380 1369 1073"> <p>Recipients MUST NOT skip this attribute if unknown.</p> <p>This flag does not mean that all IMCs and IMVs must support this attribute. Instead, any IMC or IMV that receives an attribute with this flag set to one but does not support this attribute MUST NOT process any part of the IF-M message and SHOULD return a TNC_IFM_ATTRIBUTE_NOT_SUPPORTED error in an IF-M error message.</p> <p>In order to avoid taking action on a subset of the attributes only to later find an unsupported NOSKIP flagged attribute, recipients of a multi-attribute IF-M message might need to scan all of the attributes prior to acting upon any attribute.</p> <p>When set to zero, recipients SHOULD skip any unsupported attributes and continue processing the next attribute.</p> </td> </tr> <tr> <td data-bbox="407 1073 609 1283"> Bit 1 Correlation ID (COR) </td> <td data-bbox="609 1073 1369 1283"> <p>Indicates whether the optional Correlation ID value is included in the header.</p> <p>When set to one, a 32 bit Correlation ID field is present. Otherwise when set to zero, no Correlation ID is included.</p> </td> </tr> <tr> <td data-bbox="407 1283 609 1409"> Bit 2-7 </td> <td data-bbox="609 1283 1369 1409"> <p>Reserved for future use. These bits MUST be set to 0 on transmission and ignored upon reception</p> </td> </tr> </tbody> </table>	Bit Encoding	Description	Bit 0 No Skip Flag (NOSKIP)	<p>Recipients MUST NOT skip this attribute if unknown.</p> <p>This flag does not mean that all IMCs and IMVs must support this attribute. Instead, any IMC or IMV that receives an attribute with this flag set to one but does not support this attribute MUST NOT process any part of the IF-M message and SHOULD return a TNC_IFM_ATTRIBUTE_NOT_SUPPORTED error in an IF-M error message.</p> <p>In order to avoid taking action on a subset of the attributes only to later find an unsupported NOSKIP flagged attribute, recipients of a multi-attribute IF-M message might need to scan all of the attributes prior to acting upon any attribute.</p> <p>When set to zero, recipients SHOULD skip any unsupported attributes and continue processing the next attribute.</p>	Bit 1 Correlation ID (COR)	<p>Indicates whether the optional Correlation ID value is included in the header.</p> <p>When set to one, a 32 bit Correlation ID field is present. Otherwise when set to zero, no Correlation ID is included.</p>	Bit 2-7	<p>Reserved for future use. These bits MUST be set to 0 on transmission and ignored upon reception</p>
Bit Encoding	Description								
Bit 0 No Skip Flag (NOSKIP)	<p>Recipients MUST NOT skip this attribute if unknown.</p> <p>This flag does not mean that all IMCs and IMVs must support this attribute. Instead, any IMC or IMV that receives an attribute with this flag set to one but does not support this attribute MUST NOT process any part of the IF-M message and SHOULD return a TNC_IFM_ATTRIBUTE_NOT_SUPPORTED error in an IF-M error message.</p> <p>In order to avoid taking action on a subset of the attributes only to later find an unsupported NOSKIP flagged attribute, recipients of a multi-attribute IF-M message might need to scan all of the attributes prior to acting upon any attribute.</p> <p>When set to zero, recipients SHOULD skip any unsupported attributes and continue processing the next attribute.</p>								
Bit 1 Correlation ID (COR)	<p>Indicates whether the optional Correlation ID value is included in the header.</p> <p>When set to one, a 32 bit Correlation ID field is present. Otherwise when set to zero, no Correlation ID is included.</p>								
Bit 2-7	<p>Reserved for future use. These bits MUST be set to 0 on transmission and ignored upon reception</p>								
Vendor ID	<p>This field indicates the owner of the name space associated with the Attribute Type. This is accomplished by specifying the 24 bit SMI Private Enterprise Number Vendor ID[REF] of the party who owns the Attribute Type name space. TCG standard attributes defined in this specification MUST use the TCG SMI Private Enterprise Number value (0x005597) in this field.</p>								
Attribute Type	<p>This field defines the type of the attribute within the scope of the specified vendor name space (Vendor ID) included in the Attribute Value field. The specific TNC standard values allowable in this field when the Vendor ID is the TCG SMI Private Enterprise Number value (0x005597) are defined in section 4.2.</p>								
Attribute	<p>This field contains the length in octets of the entire</p>								

Length	Attribute including the Attribute's TLV header. Implementations that do not support the specified Attribute Type can use this length to skip over the attribute to the next attribute. Note that while this field is 4 octets the maximum usable attribute length is likely to be less than 2 ³² due to limitations of the underlying protocol stack.
Correlation ID	<p>This optional field MUST only exist when the COR flag is set to one. Normally this field is not expected to exist so this field is absent. However when a single IMC responds with multiple of the same type of attribute to an IMV, this field MUST uniquely identify each repeated attribute. The Correlation ID value MUST be constant per product for an entire IF-TNCCS session so the IMV can correlate attributes requested earlier about the same product.</p> <p>An IMC that is able to produce attributes describing multiple products on an endpoint uses this field to consistently tag each attribute to facilitate correlation by the IMV. For example, if an IMV requests Product Information and Version attributes for the anti-malware component, this special IMC would produce two Product Information and two Version attributes each with a constant Correlation ID per product. The Product Information and Version attributes describing the same product would have the same Correlation ID. This allows the IMV to associate the Product Information and Version attributes that apply to a single product. Because the Product Information and Version attribute requests might be requested at different times, it is important that the IMC use a consistent value for each product it is able to report upon. For example a multi-product IMC might create a persistent table of locally unique IDs (e.g. counters) for each product it reports upon for situations where a Correlation ID is necessary.</p> <p>Note that if an endpoint has two anti-malware IMCs installed with each returning a single attribute the Correlation ID is not required. This is because the IMV can discover the identity of the originating IMC (included in the IF-TNCCS protocol) to determine which anti-malware product the attribute is associated with.</p>
Attribute Value	This field varies depending on the particular type of attribute being expressed. The contents of this field for each of the TNC standard based attributes are defined in section 4.2.

4.2 TNC Standard Attributes

This section defines the IF-M 1.0 set of TNC standard attributes. These attributes all use the TCG SMI Private Enterprise Number (0x005597) in the Vendor ID field of the IF-M Attribute Header described in section 4.1. The following table briefly describes each attribute and defines the value to be used in the Attribute Type field of the IF-M Attribute Header. Attributes that have a similar purpose are grouped (e.g. attributes that include measurement data) and are allocated attribute type values from a particular portion of

the enumeration to ease recognition by a recipient (e.g. measurement data attributes fall between 0x00000010-0x00000FFF). Later subsections provide detailed specifications for each attribute.

The following TNC standard attributes are defined in this specification:

Attribute Purpose	Attribute Name	TNC Standard Attribute Type	Description
Reserved			
	Reserved	0x00000000	Reserved for use in specification examples, experimentation and testing.
Measurement Request			
	Attribute Request	0x00000001	Contains a list of attribute type values defining the attributes desired from the IMCs
Security			
	Security Capabilities	0x00100000	See IF-M Security: Bindings to CMS [IF-M-SEC] specification for details
	CMS Error Code	0x00200000	See IF-M Security: Bindings to CMS specification for details
	CMS Protected Content	0x00300000	See IF-M Security: Bindings to CMS specification for details
Measurement Data			
	Product Information	0x00000010	Manufacturer and product information for the component
	Numeric Version	0x00000020	Detailed numeric version of the component. This structure is expected to be useful for describing the running operating system but may have other uses.
	String Version	0x00000030	String representation of the version of the particular component
	Operational Status	0x00000040	Describes whether the component is running on the endpoint

	Port Filter	0x00000050	Lists the set of ports (e.g. TCP port 80 for HTTP) that are allowed or blocked on the endpoint
	Installed Packages	0x00000060	List of software packages installed on endpoint that provide the requested component. A package is a collection of software installed as a unit. For example, an RPM on Linux or a package on Solaris.
Result			
	IMV Assessment Results	0x00010000	IMV component-level assessment results
	Remediation Instructions	0x00020000	Remediation instructions for updating the component (e.g. URI)
	IF-M Error	0x00030000	IF-M message or attribute processing error

The following subsections discuss the usage, format and semantics of the Attribute Value field for each type of TNC standard attribute. These fields follow the IF-M Attribute Header in each IF-M attribute.

4.2.1 Attribute Applicability

This section summarizes which of the TNC standard attributes are required to be support by IMC and IMV supporting each of the types of components described in this specification. IMCs and IMVs associated with a particular type of component SHOULD NOT support additional TNC standard attributes to avoid vagueness in how they are interpreted (e.g. returning a Port Filter attribute for an Anti-Virus component). However IMCs and IMVs are not required to support all the attributes applicable to their component.

The following table indicates whether an IMC or IMV claiming full support for a particular component type needs to support many of the defined standard attribute types. Every IMC and IMV SHOULD support the Attribute Request attribute allowing an IMV to request particular attributes. Similarly, every IMC and IMV SHOULD support the security related attributes defined in section 4.2 and described in the IF-M security specification. Note that the choice of whether an IMC wishes to send each type of attribute to an IMV should be under the control of local privacy policy.

	Product Info.	Numeric Version	String Version	Oper. Status	Port Filter	Installed Pkgs	IMV Assess. Results	Remed. Instruct	IF-M Error
Operating System	MUST	SHOULD	MUST	MAY	MUST NOT	SHOULD NOT [1]	SHOULD	MAY	MUST

Anti-Virus	MUST	MAY	MUST	SHOULD	MUST NOT	SHOULD	SHOULD	MAY	MUST
Anti-Spyware	MUST	MAY	MUST	SHOULD	MUST NOT	SHOULD	SHOULD	MAY	MUST
Anti-Malware	MUST	MAY	MUST	SHOULD	MUST NOT	SHOULD	SHOULD	MAY	MUST
Firewall	MUST	MAY	MUST	SHOULD	SHOULD	SHOULD	SHOULD	MAY	MUST
Intrusion Detection/ Prevent.	MUST	MAY	MUST	SHOULD	MUST NOT	SHOULD	SHOULD	MAY	MUST
Virtual Private Network	MUST	MAY	MUST	MAY	SHOULD	SHOULD	SHOULD	MAY	MUST

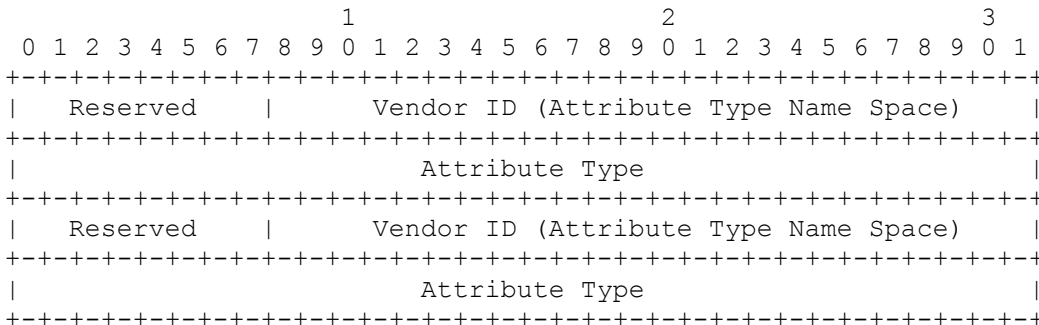
Attribute Support Requirements for each Component Type

[1] – Support for returning all the installed packages for an entire operating system is discouraged due to the size of the message and the potential resulting timeouts in underlying transports such as 802.1X.

4.2.2 Attribute Request

The Attribute Request allows an IMV to request certain attributes from the registered set of IMCs. The registered IMCs MAY choose to send all, a subset or none of the request attributes but MUST NOT send attributes that were not requested (except error attributes). Each Attribute Request MUST contain at least one vendor-defined or TNC standard attribute type. Because the length of a Vendor ID paired with an Attribute Type has a fixed length of 8 octets, the number of requested attributes can be computed using the Attribute Length field (in the Attribute Header).

For the Attribute Request attribute, the IF-M Attribute Header's Vendor ID MUST be set to the TCG SMI Private Enterprise Number (0x005597) and Attribute Type MUST be set to 0x00000001. The Attribute Value contains the following information:



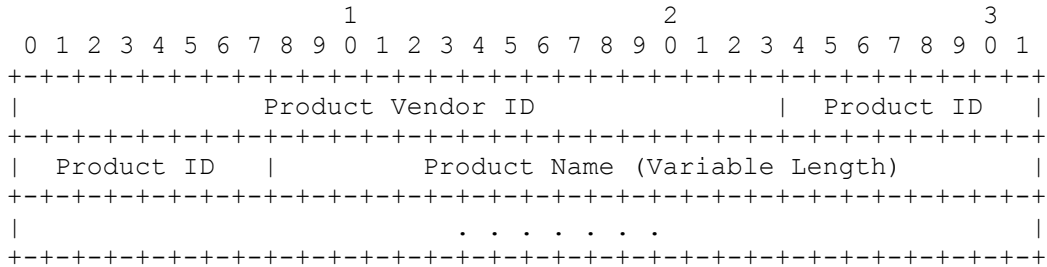
Header Field	Description
Reserved	Reserved for future use. This field MUST be set to zero on transmission and ignored upon reception.
Vendor ID (Attribute Type Name Space)	The SMI Private Enterprise Number of the vendor who controls the name space for the following Attribute Type. This field enables vendor and standards based attributes to be used without potential collisions.

	Any TNC standard attributes defined in section 4.2 MUST use the TCG SMI Private Enterprise Number (0x005597) in this field. Vendor-defined attributes MUST use the SMI Private Enterprise Number of the vendor who defined the attribute.
Attribute Type	Each attribute type field indicates the specific attribute requested. The TNC standard Attribute Types defined in section 4.2 that have a security or measurement data related purpose can be requested using this field. Other attribute purposes from the TNC standard attribute name space MUST NOT be included in this field (e.g. requesting an IF-M Error attribute). All other attribute type values not defined in this specification MUST be used within a vendor-defined name space.

4.2.3 Product Information

This attribute contains product level information about the product that implements the component specified in the component type field in the IF-TNCCS header (see section 3.2) of a TNC standard IF-M message. For example, if the component type is anti-virus, this attribute would contain information about the anti-virus product installed on the endpoint.

This attribute includes both vendor and product level identification information. For the Product Information attribute, the IF-M Attribute Header's Vendor ID MUST be set to the TCG SMI Private Enterprise Number (0x005597) and Attribute Type MUST be set to 0x00000010. The Attribute Value contains the following information:

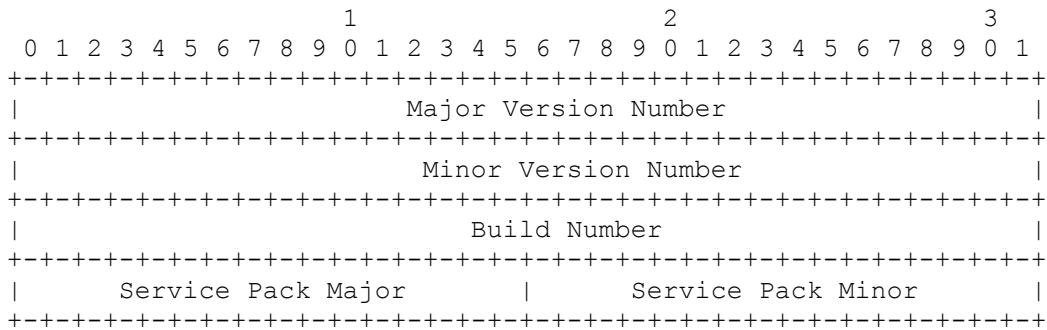


Header Field	Description
Product Vendor ID	This field contains the IANA assigned SMI Private Enterprise Number for the organization that created the product. If the product creator does not have an SMI Private Enterprise Number or is unknown, this value MUST be set to 0xffffffff.
Product ID	This enumeration uniquely identifies the product containing the requested component from the product's vendor. If the product vendor is unknown, the Product ID field MUST be 0. This field is a vendor-defined field to identify the particular component present on the endpoint. For example, Symantec offers numerous anti-virus oriented products. If the request was for an anti-virus component, this enumeration could be used to identify which anti-virus product is present on the system.

Product Name	<p>This variable length field contains a UTF-8 string identifying the product (e.g. "Symantec Norton AntiVirus™ 2008") in enough detail to unambiguously distinguish it from other products from the vendor. This might require inclusion of information about the edition or other product marketing information to assure it is unambiguously identified. Products associated with a known vendor who does not have a registered SMI Private Enterprise Number SHOULD be represented using a combination of the vendor name and full product name (e.g. "Ubuntu® IPTables" for the IPTables firewall in the Ubuntu distribution of Linux).</p> <p>The length of this field can be determined by starting with the Attribute Length field in the attribute header and subtracting the size of the fixed length fields that precede it. However, implementers should be careful that the Attribute Length is not less than the size of the fixed length fields. Such a circumstance could cause a buffer overflow if not handled properly. It is a syntax error and should result in a TNC_IFM_MALFORMED_MESSAGE error code.</p>
--------------	--

4.2.4 Numeric Version

This attribute describes the detailed version information about the requested component (e.g. operating system) in use on the endpoint. This version includes structured values for the version information to enable IMVs to perform comparative operations on the version. The version information included is associated with a particular product, so IMV are expected to also possess the corresponding Product Information attribute when interpreting this attribute. Some IMC may not be able to determine some or all of this information for its component. Similarly many components do not use such granular version information. It's envisioned that this attribute could be useful for describing the version of the operating system. For the Numeric Version attribute, the IF-M Attribute Header's Vendor ID MUST be set to the TCG SMI Private Enterprise Number (0x005597) and Attribute Type MUST be set to 0x0000020. The Attribute Value contains the following information:

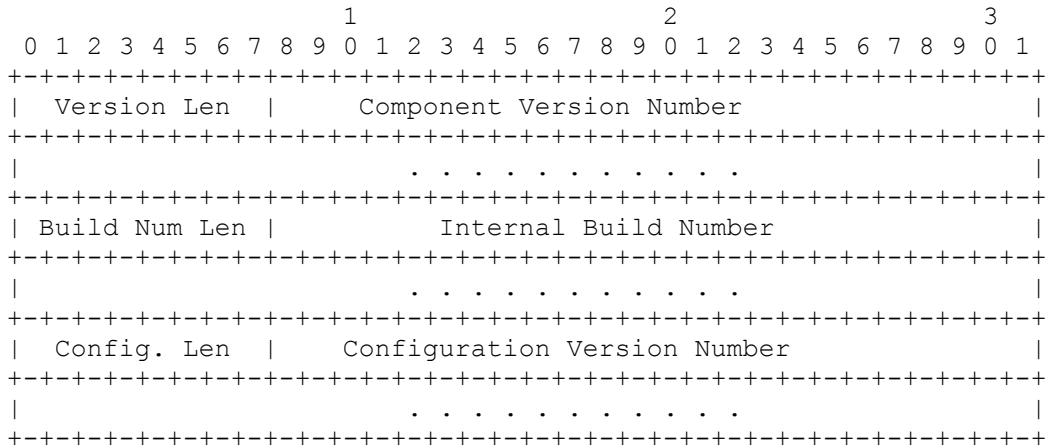


Header Field	Description
Major Version Number	This field contains the major version number for the component (e.g. Windows® Vista is 6). For operating systems, this value can be obtained using APIs like GetVersionEx on Windows and uname on Solaris.

Minor Version Number	This field contains the minor version number for the component (e.g. Solaris™ 10 is 10). For operating systems, this value can be obtained using APIs like uname on Linux and oslevel on AIX.
Build Number	This field contains the internal engineering group's build number. This provides more granularity than the minor version number as many builds might occur leading up to an official release major/minor version. For operating systems, this value can be obtained using APIs like GetVersionEX on Windows and uname on Linux. If this field is not used, the value MUST be 0.
Service Pack Major	If applicable, this field contains the service pack major version number as provided by an API like GetVersionEX on Windows. If this field is not used, the value MUST be 0.
Service Pack Minor	If applicable, this field corresponds to the Service Pack Major above but provides more granularity into the service pack version. If this field is not used, the value MUST be 0.

4.2.5 String Version

This attribute contains the version information of the component defined in the component type field in the IF-TNCCS header (see section 3.2) for a TNC standard IF-M message. The version information included is associated with a particular product, so IMVs are expected to also possess the corresponding Product Information attribute when interpreting this attribute. For the String Version attribute, the IF-M Attribute Header's Vendor ID MUST be set to the TCG SMI Private Enterprise Number (0x005597) and Attribute Type MUST be set to 0x00000030. The Attribute Value contains the following information:

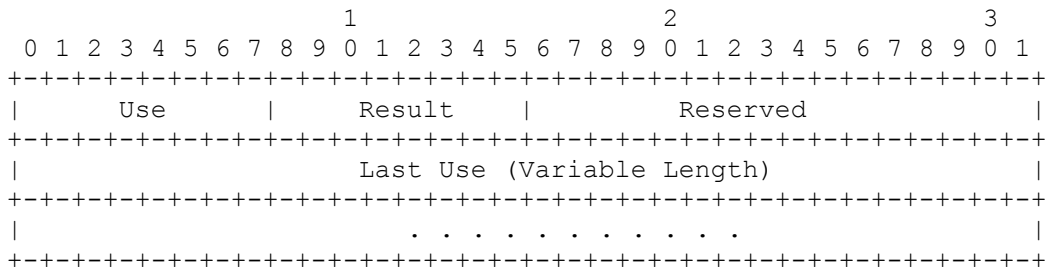


Header Field	Description
Version Len	This field defines the number of octets in the Component Version Number string field.
Component	This field contains a UTF-8 string identifying the version

Version Number	of the component (e.g. "1.12.23.114"). This field MUST be sized to fit the version string and MUST NOT include extra octets for padding or NUL character termination. Various products use a wide range of different formats for version strings. Some use alphabetic characters, white space, and punctuation. Therefore, the syntax and semantics of this version string are not defined.
Build Num Len	This field defines the number of octets in the Internal Build Number string field. For components where the internal build number is unavailable or unknown, this field MUST be set to zero and the Internal Build Number is not present.
Internal Build Number	This field contains a UTF-8 string representing the vendor internal engineering build number of the product. In some cases this value is used to differentiate different minor (or test) releases of a product prior to declaring a new official version release.
Config. Len	This field defines the number of octets in the Configuration Version Number string field.
Configuration Version Number	This field contains a UTF-8 string identifying the version of the configuration used by the component. This version SHOULD represent the overall configuration version even if several configuration policy files or settings are used. Vendors MAY include multiple versions if a single version is not practical. This field MUST be sized to fit the version string and MUST NOT include extra octets for padding or NUL character termination. Various products use a wide range of different formats for version strings. Some use alphabetic characters, white space, and punctuation. Therefore, the syntax and semantics of this version string are not defined.

4.2.6 Operational Status

This attribute describes the operational status of the component defined in the component type field in the IF-TNCCS header (see section 3.2). For the Operational Status attribute, the IF-M Attribute Header's Vendor ID MUST be set to the TCG SMI Private Enterprise Number (0x005597) and Attribute Type MUST be set to 0x00000040. The Attribute Value contains the following information:

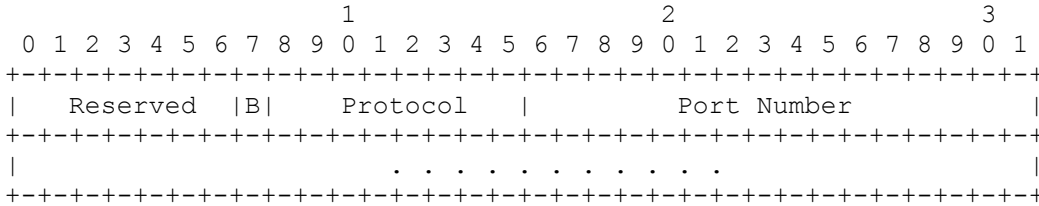


Header Field	Description												
Use	<p>Operational status of the usability of the component.</p> <table border="1" data-bbox="397 321 1367 516"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Status is unknown or other</td> </tr> <tr> <td>1</td> <td>Not installed on system</td> </tr> <tr> <td>2</td> <td>Installed but not operational</td> </tr> <tr> <td>3</td> <td>Operational</td> </tr> <tr> <td>4+</td> <td>Reserved for future use</td> </tr> </tbody> </table>	Value	Description	0	Status is unknown or other	1	Not installed on system	2	Installed but not operational	3	Operational	4+	Reserved for future use
Value	Description												
0	Status is unknown or other												
1	Not installed on system												
2	Installed but not operational												
3	Operational												
4+	Reserved for future use												
Result	<p>This field contains the result of the last use of an operational component. This field MUST be set to zero when the Use field contains a value other than Operational (3). The following table enumerates the values of this field:</p> <table border="1" data-bbox="397 720 1367 915"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Status is unknown or other</td> </tr> <tr> <td>1</td> <td>Successful use with no errors detected</td> </tr> <tr> <td>2</td> <td>Successful use with an error detected</td> </tr> <tr> <td>3</td> <td>Last use aborted or otherwise unsuccessful</td> </tr> <tr> <td>4+</td> <td>Reserved for future use</td> </tr> </tbody> </table>	Value	Description	0	Status is unknown or other	1	Successful use with no errors detected	2	Successful use with an error detected	3	Last use aborted or otherwise unsuccessful	4+	Reserved for future use
Value	Description												
0	Status is unknown or other												
1	Successful use with no errors detected												
2	Successful use with an error detected												
3	Last use aborted or otherwise unsuccessful												
4+	Reserved for future use												
Reserved	<p>Reserved for future use. This field MUST be set to zero on transmission and ignored upon reception.</p>												
Last Use	<p>This field contains the date and time of the last use of the component, if known. The Last Use date and time MUST be represented as an RFC 3339[RFC3339] compliant ASCII string in Coordinated Universal Time (UTC) time with the additional restrictions that the 't' delimiter and the 'z' suffix MUST be capitalized and fractional seconds (time-secfrac) MUST NOT be included. The last use string MUST NOT be NUL terminated. If the last use is not known, this field MUST contain "0000-00-00T00:00:00Z" allowing this attribute to be fixed length. Note that this reserved value is not RFC 3339 compliant (zero month).</p> <p>This encoding produces an easy to read, parse and interpret string in YYYY-MM-DDTHH:MM:SSZ format that can precisely define a particular second in UTC time. For example, 9:05:00AM EST on January 19, 1995 can be represented as "1995-01-19T14:05:00Z".</p>												

4.2.7 Port Filter

This attribute includes the list of port numbers and their associated protocols (e.g. TCP and UDP) that are currently blocked or allowed by the host-based firewall on the endpoint. Each Protocol and Port Number combination uses 4 octets, so the number of filtered ports can be calculated using the Attribute Length in the Attribute Header (see section 4.1).

For the Port Filter attribute, the IF-M Attribute Header's Vendor ID MUST be set to the TCG SMI Private Enterprise Number (0x005597) and Attribute Type MUST be set to 0x00000050. The Attribute Value contains the following information:

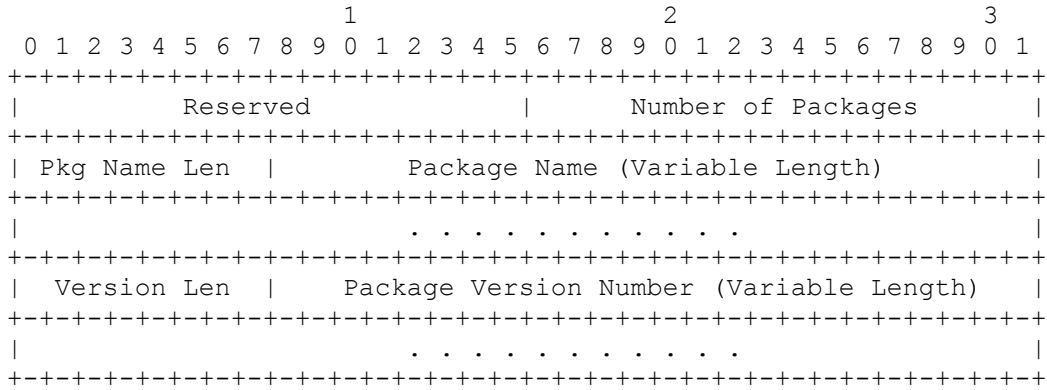


Header Field	Description
Reserved	Reserved for future use. This field MUST be set to zero on transmission and ignored upon reception.
B Flag (Blocked or Allowed Port)	<p>This single bit field indicates whether the following port is blocked or allowed. This bit MUST be set to one if the protocol/port combination is blocked otherwise this field MUST be set to zero. This field was provided to allow for more abbreviated reporting of the port filtering policy (e.g. when all ports are blocked except a few, this could just list the few as not blocked).</p> <p>IMCs MUST NOT provide a mixed list of block and non-blocked ports for a particular protocol. IMCs MUST NOT list the same Protocol and Port Number combination twice in an attribute. IMCs MAY list all blocked ports for one protocol and all allowed ports for a different protocol in this attribute using the B flag to indicate whether each are blocked.</p> <p>For example, an IMC might list all the blocked TCP ports but only list the allowed UDP ports. However it MUST NOT list some blocked TCP ports and some other allowed TCP ports.</p>
Protocol	This field mirrors the IPv4 Protocol and IPv6 Next Header field. The allowable values for this field are managed by the IANA. The current list of protocols can be found at: http://www.iana.org/assignments/protocol-numbers
Port Number	This field mirrors the port allocation space assigned for the above specified Protocol field. For example, if the Protocol is TCP (0x0006) then the port numbers are those associated with TCP. The TCP and UDP port number assignments are managed by the IANA and can be found at: http://www.iana.org/assignments/port-numbers

4.2.8 Installed Packages

This attribute contains meta-data about installed packages that comprise a particular component. This allows an IMV to check the versions of packages that are installed for a particular component. For the Installed Packages attribute, the IF-M Attribute Header's Vendor ID MUST be set to the TCG SMI Private

Enterprise Number (0x005597) and Attribute Type MUST be set to 0x00000060. The Attribute Value contains the following information:

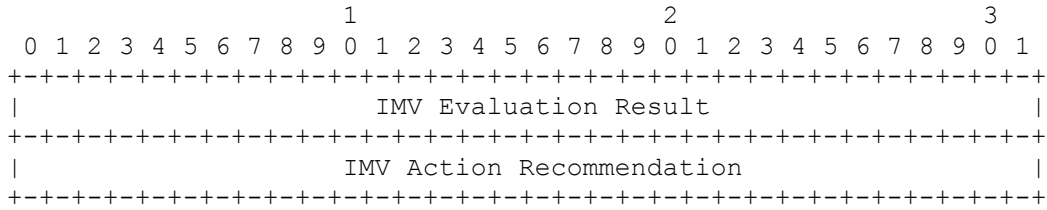


Header Field	Description
Reserved	Reserved for future use. This field MUST be set to zero on transmission and ignored upon reception.
Number of Packages	This field indicates the number of packages described in this attribute. Each package description includes both the variable length package name and its version as these are always both present.
Pkg Name Len	This field indicates the length of the Package Name in octets.
Package Name	This field contains a UTF-8 string identifying the name of the package associated with the type of component. This field MUST be sized to fit the version string and MUST NOT include extra octets for padding or NUL character termination. This name is package technology dependent (e.g. RPM names on Linux).
Version Len	This field indicates the length of the Package Version Number in octets.
Package Version Number	This field contains a UTF-8 string identifying the version (e.g. "1.2.3.4") of the package named by the prior field in this attribute. This field MUST be sized to fit the version string and MUST NOT include extra octets for padding or NUL character termination.

4.2.9 IMV Assessment Results

This attribute contains the final assessment result and action recommendation from a particular IMV. This value might be returned to an IMC for information purposes (e.g. when an assessment is successful) or in conjunction with other attributes indicating that corrective action is required. For the IMV Assessment Results attribute, the IF-M Attribute Header's Vendor ID MUST be set to the TCG SMI Private Enterprise

Number (0x005597) and Attribute Type MUST be set to 0x00010000. The Attribute Value contains the following information:



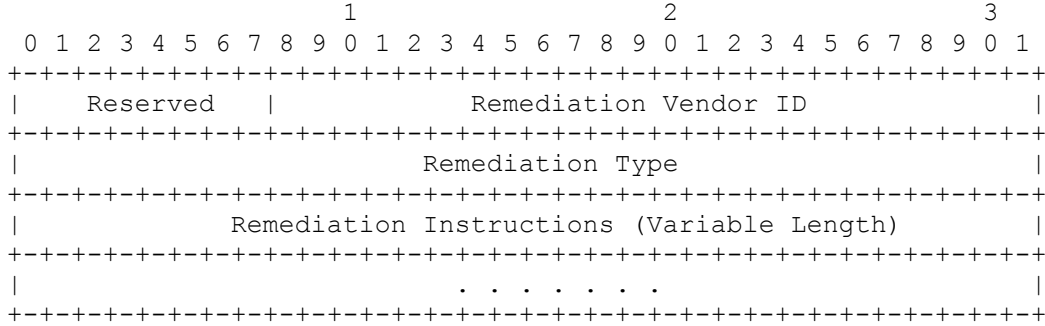
Header Field	Description												
IMV Evaluation Result	<p>This field contains the numeric evaluation result returned by the IMV to the TNCs. This field MUST take one of the TNC standard values from the following table:</p> <table border="1" data-bbox="443 684 1377 1150"> <thead> <tr> <th data-bbox="443 684 565 716">Value</th> <th data-bbox="565 684 1377 716">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="443 716 565 779">0</td> <td data-bbox="565 716 1377 779">IMV's component was assessed to be compliant with policy (TNC IFM EVALUATION RESULT COMPLIANT)</td> </tr> <tr> <td data-bbox="443 779 565 905">1</td> <td data-bbox="565 779 1377 905">IMV's component was assessed to be non-compliant with policy but the difference from compliance was minor. (TNC IFM EVALUATION RESULT NON COMPLIANT MINOR)</td> </tr> <tr> <td data-bbox="443 905 565 999">2</td> <td data-bbox="565 905 1377 999">IMV's component was assessed to be non-compliant and the assessed difference was very significant. (TNC IFM EVALUATION RESULT NON COMPLIANT MAJOR)</td> </tr> <tr> <td data-bbox="443 999 565 1062">3</td> <td data-bbox="565 999 1377 1062">IMV was unable to determine policy compliance due to an error (TNC IFM EVALUATION RESULT ERROR)</td> </tr> <tr> <td data-bbox="443 1062 565 1150">4</td> <td data-bbox="565 1062 1377 1150">IMV was unable to determine whether IMC measurements are compliant with policy. (TNC IFM EVALUATION RESULT DONT KNOW)</td> </tr> </tbody> </table>	Value	Description	0	IMV's component was assessed to be compliant with policy (TNC IFM EVALUATION RESULT COMPLIANT)	1	IMV's component was assessed to be non-compliant with policy but the difference from compliance was minor. (TNC IFM EVALUATION RESULT NON COMPLIANT MINOR)	2	IMV's component was assessed to be non-compliant and the assessed difference was very significant. (TNC IFM EVALUATION RESULT NON COMPLIANT MAJOR)	3	IMV was unable to determine policy compliance due to an error (TNC IFM EVALUATION RESULT ERROR)	4	IMV was unable to determine whether IMC measurements are compliant with policy. (TNC IFM EVALUATION RESULT DONT KNOW)
Value	Description												
0	IMV's component was assessed to be compliant with policy (TNC IFM EVALUATION RESULT COMPLIANT)												
1	IMV's component was assessed to be non-compliant with policy but the difference from compliance was minor. (TNC IFM EVALUATION RESULT NON COMPLIANT MINOR)												
2	IMV's component was assessed to be non-compliant and the assessed difference was very significant. (TNC IFM EVALUATION RESULT NON COMPLIANT MAJOR)												
3	IMV was unable to determine policy compliance due to an error (TNC IFM EVALUATION RESULT ERROR)												
4	IMV was unable to determine whether IMC measurements are compliant with policy. (TNC IFM EVALUATION RESULT DONT KNOW)												
IMV Action Recommendation	<p>This field contains the numeric action recommendation returned from the IMV to the TNCs. This field MUST take one of the TNC standard values from the following table:</p> <table border="1" data-bbox="443 1283 1377 1629"> <thead> <tr> <th data-bbox="443 1283 565 1314">Value</th> <th data-bbox="565 1283 1377 1314">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="443 1314 565 1377">0</td> <td data-bbox="565 1314 1377 1377">IMV recommends allowing access (TNC IFM ACTION RECOMMENDATION ALLOW)</td> </tr> <tr> <td data-bbox="443 1377 565 1440">1</td> <td data-bbox="565 1377 1377 1440">IMV recommends not to allow access (TNC IFM ACTION RECOMMENDATION NO ACCESS)</td> </tr> <tr> <td data-bbox="443 1440 565 1535">2</td> <td data-bbox="565 1440 1377 1535">IMV recommends limited access. This access may be expanded after remediation (TNC IFM ACTION RECOMMENDATION ISOLATE)</td> </tr> <tr> <td data-bbox="443 1535 565 1598">3</td> <td data-bbox="565 1535 1377 1598">IMV was unable to make a recommendation (TNC IFM ACTION RECOMMENDATION NONE)</td> </tr> <tr> <td data-bbox="443 1598 565 1629">4+</td> <td data-bbox="565 1598 1377 1629">Reserved for future use</td> </tr> </tbody> </table>	Value	Description	0	IMV recommends allowing access (TNC IFM ACTION RECOMMENDATION ALLOW)	1	IMV recommends not to allow access (TNC IFM ACTION RECOMMENDATION NO ACCESS)	2	IMV recommends limited access. This access may be expanded after remediation (TNC IFM ACTION RECOMMENDATION ISOLATE)	3	IMV was unable to make a recommendation (TNC IFM ACTION RECOMMENDATION NONE)	4+	Reserved for future use
Value	Description												
0	IMV recommends allowing access (TNC IFM ACTION RECOMMENDATION ALLOW)												
1	IMV recommends not to allow access (TNC IFM ACTION RECOMMENDATION NO ACCESS)												
2	IMV recommends limited access. This access may be expanded after remediation (TNC IFM ACTION RECOMMENDATION ISOLATE)												
3	IMV was unable to make a recommendation (TNC IFM ACTION RECOMMENDATION NONE)												
4+	Reserved for future use												

4.2.10 Remediation Instructions

This attribute contains remediation instructions for updating a component that has not passed an evaluation. Because many remediation approaches exist, one goal for this attribute was to create a single extensible attribute type capable of carrying instructions for a wide variety of remediation techniques. This

single attribute type approach also eases the TNCC or TNCS's ability to recognize and potentially block when an IMV is requesting an IMC remediate a particular component in case this wasn't desired.

For the Remediation Instructions attribute, the IF-M Attribute Header's Vendor ID **MUST** be set to the TCG SMI Private Enterprise Number (0x005597) and Attribute Type **MUST** be set to 0x00020000. The Attribute Value contains the following information:



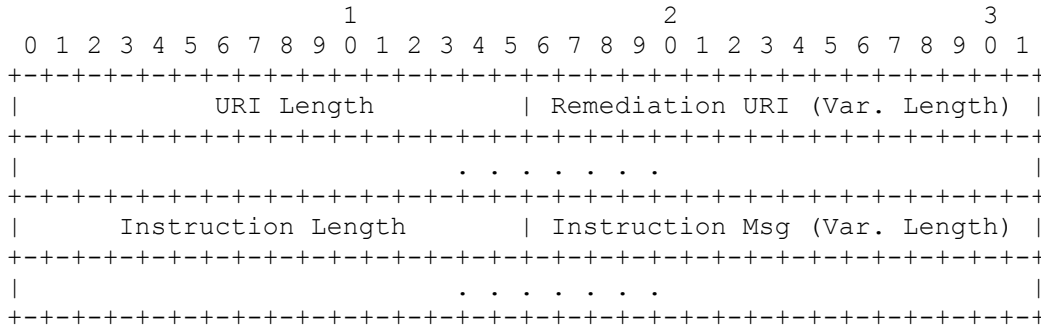
Header Field	Description								
Reserved	Reserved for future use. This field MUST be set to zero on transmission and ignored upon reception.								
Remediation Vendor ID	This field contains the IANA assigned SMI Private Enterprise Number for the vendor whose Remediation Type name space is being used in the attribute. For TCG standards based Remediation Type values this field MUST be set to 0x005597. For other vendor-defined types of remediation, this field MUST contain the vendor's SMI Private Enterprise Number.								
Remediation Type	This field identifies the format and semantics of remediation instructions used within the attribute. This type exists within the scope of Vendor ID defined name space allowing for both vendor-defined and TCG standard name spaces. When the Vendor ID is set to the TCG Private Enterprise Number, the following table lists the supported Remediation Type values: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Invalid value (MUST NOT be used)</td> </tr> <tr> <td>1</td> <td>TNC URI-Based Remediation</td> </tr> <tr> <td>2+</td> <td>Reserved for future use</td> </tr> </tbody> </table>	Value	Description	0	Invalid value (MUST NOT be used)	1	TNC URI-Based Remediation	2+	Reserved for future use
Value	Description								
0	Invalid value (MUST NOT be used)								
1	TNC URI-Based Remediation								
2+	Reserved for future use								
Remediation Instructions	This field varies depending on the particular type of remediation instructions being expressed. The contents of this field for each of the TNC standard based remediation instructions are defined in section 4.2.10.2.								

4.2.10.1 TNC Remediation Instructions

The following subsection shows the TCG standard format that **MUST** be used in the Remediation Instructions field when the Remediation Vendor ID is set to the TCG SMI Private Enterprise Number. Additional TNC standard remediation instruction types are envisioned to be added in future revisions of this specification.

4.2.10.2 TNC URI-Based Remediation

This attribute provides information to facilitate a TNC standard, semi-manual remediation where a human could be required to take a corrective action using the provided URI to the remediation server. The Attribute Value contains the following information:



Header Field	Description
URI Length	This field defines the length in octets of the Remediation URI variable length field. A length of 0 indicates no Remediation URI is included.
Remediation URI	This field contains a URI referencing the service capable of providing the remediation updates to the system. This URI MUST be converted to a UTF-8 sequence of octets and then percent encoded where necessary [RFC3986].
Instruction Length	This field defines the length in octets of the Instruction Message variable length field. A length of 0 indicates no Instruction Message is included.
Instruction Msg	This variable length UTF-8 string contains a message for the user explaining how to use the remediation URI to update the system.

4.2.11 IF-M Error

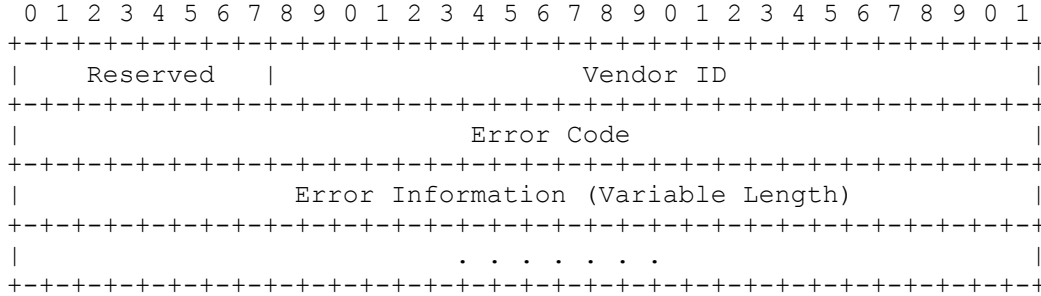
This attribute contains error codes and supplemental information regarding IF-M level messaging errors. For the IF-M Error attribute, the IF-M Attribute Header’s Vendor ID MUST be set to the TCG SMI Private Enterprise Number (0x005597) and Attribute Type MUST be set to 0x00030000.

An IF-M error SHOULD be sent with the same IF-M Vendor ID and IF-M Subtype used by the IF-M message that caused the error so that the error code is sent to the party who sent the offending IF-M message. Other measures (such as setting IF-TNCCS’s EXCL flag and the IMC Identifier or IMV Identifier fields) SHOULD also be taken to attempt to ensure that only the party who sent the offending message receives the error.

When an IF-M error is received, the recipient MUST NOT respond with an IF-M error because this could result in an infinite loop of errors. Instead, the recipient MAY log the error, modify its behavior to attempt to avoid the error (attempting to avoid loops or long strings of errors), ignore the error, terminate the assessment, or take other action as appropriate (as long as it is consistent with the requirements of this specification).

The Attribute Value contains the following information:





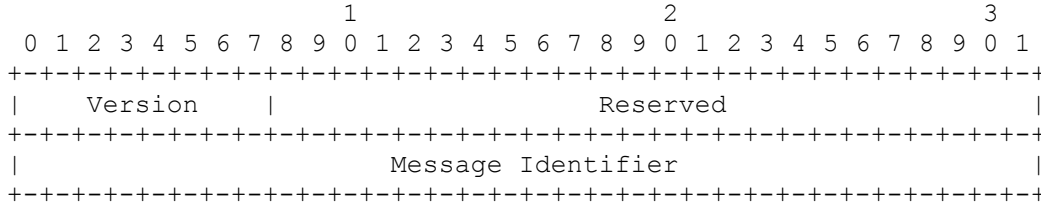
Header Field	Description										
Reserved	Reserved for future use. This field MUST be set to zero on transmission and ignored upon reception.										
Vendor ID	This field contains the IANA assigned SMI Private Enterprise Number for the vendor whose Error Code name space is being used in the attribute. For TCG standard Error Code values this field MUST be set to 0x005597. For other vendor-defined Error Code name spaces this field MUST be set to the SMI Private Enterprise Number of the vendor.										
Error Code	<p>This field contains the error code. This code exists within the scope of Vendor ID defined name space allowing for both vendor-defined and TCG standard name spaces. When the Vendor ID is set to the TCG Private Enterprise Number, the following table lists the supported TCG standard numeric error codes:</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Reserved value</td> </tr> <tr> <td>1</td> <td>IF-M attribute unknown or unsupported (TNC IFM MALFORMED MESSAGE)</td> </tr> <tr> <td>2</td> <td>IF-M protocol version not supported (TNC IFM VERSION NOT SUPPORTED)</td> </tr> <tr> <td>3</td> <td>IF-M attribute unknown or not supported (TNC_IFM_ATTRIBUTE_NOT_SUPPORTED). This can be used when an IF-M security protected attribute is received but IF-M security is not supported.</td> </tr> </tbody> </table>	Value	Description	0	Reserved value	1	IF-M attribute unknown or unsupported (TNC IFM MALFORMED MESSAGE)	2	IF-M protocol version not supported (TNC IFM VERSION NOT SUPPORTED)	3	IF-M attribute unknown or not supported (TNC_IFM_ATTRIBUTE_NOT_SUPPORTED). This can be used when an IF-M security protected attribute is received but IF-M security is not supported.
Value	Description										
0	Reserved value										
1	IF-M attribute unknown or unsupported (TNC IFM MALFORMED MESSAGE)										
2	IF-M protocol version not supported (TNC IFM VERSION NOT SUPPORTED)										
3	IF-M attribute unknown or not supported (TNC_IFM_ATTRIBUTE_NOT_SUPPORTED). This can be used when an IF-M security protected attribute is received but IF-M security is not supported.										
Error Information	<p>This variable length value provides additional context for the error. The length of this field can be determined by the recipient using the IF-M header length field.</p> <p>Subsections under 4.2.11 shows the supplemental error information that MUST be included for each TCG standard error code. This information frequently involves sending a portion of the original IF-M message so the recipient can determine which message caused the error and the messages content.</p>										

4.2.11.1 IF-M Error Structures

The following subsections show the supplemental error information that **MUST** be included in the Error Information field for each TCG standard error code. This information frequently involves sending a portion of the original IF-M message so the recipient can determine which message caused the error and the messages content.

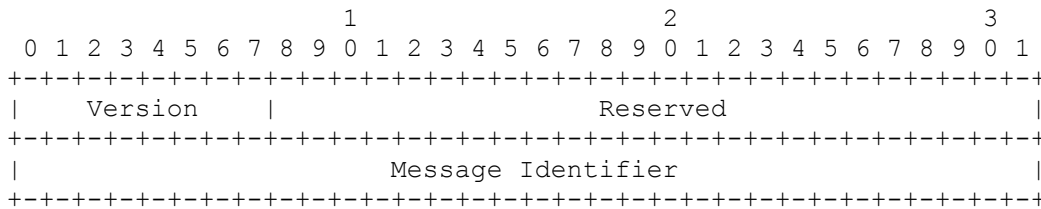
4.2.11.2 IF-M TNC_IFM_NO_ERROR Information

This field **MAY** contain the initial 8 octets of the IF-M Message Header. This field is not normally used but **MAY** be used for debugging. Recipients of this type of error **SHOULD** ignore it.



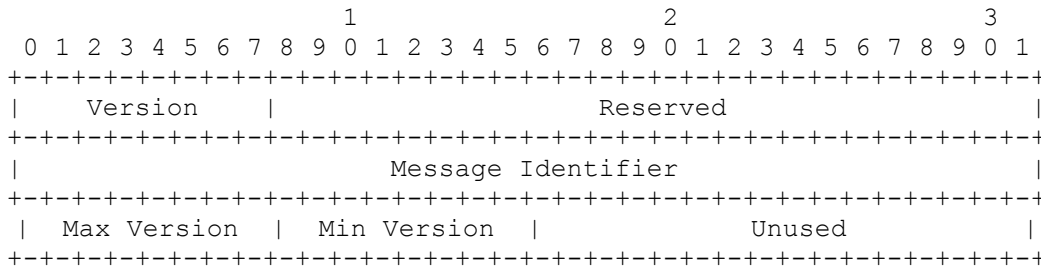
4.2.11.3 IF-M TNC_IFM_MALFORMED_MESSAGE Information

This field **MUST** contain the initial 8 octets of the IF-M Message Header that cause the error.



4.2.11.4 IF-M TNC_IFM_VERSION_NOT_SUPPORTED Information

This field **MUST** contain the initial 8 octets of the IF-M Message Header from the message that caused the error followed by the supported version information.

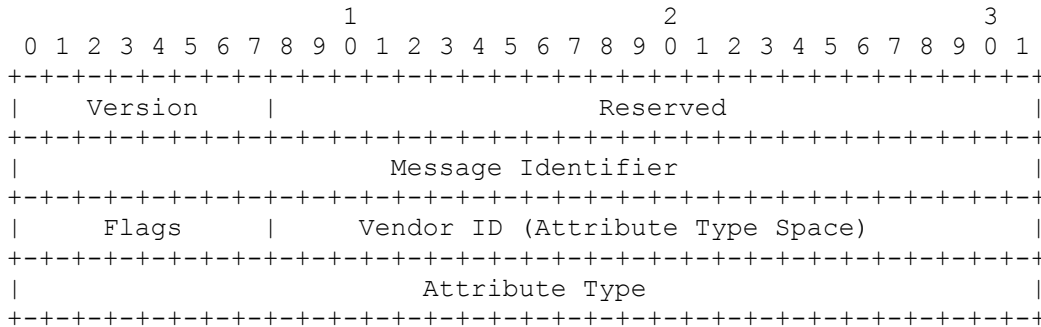


The sender of an IF-M Error message **MUST** set the Max and Min Version fields to reflect the inclusive range of IF-M protocol versions supported by the sender. The value of each version follows the description provided for the IF-M Version field described in section 3.4. The Unused field **MUST** be set to zero upon transmission and ignored upon reception. Note that the Unused field is normally called Reserved in this specification, but due to the existence of the Reserved field copied from the original message causing the error, the Unused field was given a different name to be unambiguous.

When possible, recipients of this error message **SHOULD** send future messages to the IMC or IMV that originated the error message with an IF-M protocol version within the stated range. This error message can be used to provide a basic version discovery capability between IMC and IMV.

4.2.11.5 IF-M TNC_IFM_ATTRIBUTE_NOT_SUPPORTED Information

This field MUST contain the initial 8 octets of the IF-M Message Header followed by the initial 8 octets of the IF-M attribute header for the attribute that was not supported.



4.3 Vendor-Defined Attributes

This section discusses the use of vendor-defined attributes within IF-M. The IF-M protocol was designed to allow for vendor-defined attributes to be used as a replacement where a standard attribute could be used. In some cases even the standard attributes allow for vendor-defined information to be included. It is envisioned that over time as particular vendor-defined attributes become popular, an equivalent standard attribute could be added allowing for broader interoperability.

This specification does not define vendor-defined attributes but rather highlights how such attributes can be used with IF-M without the potential for name space collisions or misinterpretations. In order to avoid collisions, IF-M uses the well-established SMI Private Enterprise Numbers as Vendor IDs to define separate name spaces for important fields within the message. For example, to ensure the uniqueness of message types while providing for vendor extensions, vendor-defined message types include the vendor’s unique Vendor ID to indicate the intended name space for the message subtype followed by the message subtype. Message types and attribute types standardized by the TCG will use the TCG’s SMI Private Enterprise Number in the Vendor ID.

SMI Private Enterprise Numbers are used to provide a separate identifier space for each vendor. IANA provides a registry for SMI Private Enterprise Numbers at <http://www.iana.org/assignments/enterprise-numbers>. Any organization (including non-profit organizations, governmental bodies, etc.) can obtain one of these numbers at no charge and thousands of organizations have done so. Within this document, SMI Private Enterprise Numbers are known as “vendor IDs”.

5 Security Considerations

This section discusses the major types of potential security threats relevant to the IF-M message protocol and summarizes the expected security protections that should be offered by IF-M security protocol(s). IF-M security protocol(s) are described in separate specifications which layer upon the base IF-M protocol described in this specification. It is envisioned that additional attribute types will be defined to facilitate the exchange of security capabilities, keys, and security protected attributes. Ultimately, the TNC deployer decides whether each particular security protection is necessary for a particular deployment environment, so the expected security protections discussed in this section highlight the need for IF-M security protocol implementations to be capable of offering the feature.

5.1 Trust Relationships

In order to understand where security countermeasures are necessary, this section starts with a discussion of where the TNC architecture envisions some trust relationships between the processing elements of the IF-M protocol. Some deployments may wish to reduce the amount of assumed trust by using an IF-M security protocol to protect the IF-M messages. The following sub-sections discuss the trust properties associated with each portion of the TNC architecture directly involved with the processing of the IF-M protocol.

5.1.1 IMC

The IMCs are trusted by IMVs to:

- Collect valid information about the component type associated with the IMC
- Report upon collected information consistent with local security and privacy policies
- Accurately report information associated with the type of component for the IF-M message
- Not act maliciously including not launching denial of service attacks against the IMVs
- Perform specified remediation instructions only when appropriate for IMC's specific product

5.1.2 IMV

The IMVs are trusted by IMCs to:

- Only request information necessary to assess the security state of the endpoint
- Make assessment decisions based on deployer defined policies
- Return the correct IMV Action Recommendation to the TNCS and when necessary the IMCs
- Discard collected information consistent with its data retention and privacy policies
- Provide accurate remediation instructions to involved IMCs when required
- Not act maliciously to TNCS and IMCs including not launching denial of service attacks against their operation

5.1.3 TNCC, TNCS and IF-TNCCS

The TNCC and TNCS are trusted by the IMC and IMV to:

- Provide a reliable transport for IF-M messages
- Deliver messages for a particular component type only to those IMCs and IMVs that have registered for them
- Not disclose any provided attributes to parties outside of the TNC assessment
- Not act maliciously to drop, duplicate or flood the IMCs and IMVs with unnecessary messages

- Not to observe, fabricate or alter the contents of an IF-M message (this trust could be minimized with an IF-M security protocol)
- Properly expose the identity of the peer TNCC or TNCS for use by IMC to make policy decisions

5.2 Security Threats

Beyond the trusted relationships assumed in section 5.1, the IF-M protocol faces a number of potential security attacks that could require targeted security countermeasures. IF-M security protocol specification(s) MUST state if and how the security protocol will safeguard against these types of attack.

Generally the IF-M protocol without the presence of security countermeasures relies upon the underlying IF-T protocol to protect the messages from attack when traveling over the network. Once the message resides on the TNCC or TNCS, it is trusted to be properly and safely delivered to the appropriate IMCs and IMVs. However in some deployments the IF-M message need to travel over network hops that are not protected by IF-T or require more assurance that only the appropriate IMC or IMV has received the message. In these cases, end to end IF-M message protection might be required. The following sub-sections focus on the potential threats where end to end protection might be desired and thus when the use of the IF-M security protocol becomes beneficial.

5.2.1 Attribute Theft

When IF-M messages are sent over unprotected network links or spanning less trusted local software stacks, the contents of the IF-M messages may be subject to information theft by an intermediary party. This theft could result in information being recorded for future use or analysis by the adversary. Attributes observed by eavesdroppers could contain information that exposes potential weaknesses in the security of the endpoint or system fingerprinting information easing the ability of the attacker to employ attacks more likely to be successful against the endpoint. The eavesdropper might also learn information about the endpoint or network policies that either singularly or collectively is considered sensitive information (e.g. certain endpoints are lacking patches or particular sub-networks have more lenient policies). IF-M attributes are not intended to carry privacy sensitive information, but should some exist in a message the adversary could come into possession of the information which could be used for other financial gain.

5.2.2 Message Fabrication

Attackers on the network or present within the TNC architecture stack could introduce fabricated IF-M messages intending to trick or cause a denial of service for aspects of an assessment. This could occur if an active attacker could launch a man-in-the-middle (MiTM) attack by proxying the IF-M messages and was able to replace undesired messages with ones easing future attack upon the endpoint. For example if IF-T security protection is not used and the TNCS proxies all assessment traffic to a remote TNCS, the proxy could eavesdrop and replace the IMV assessment results attribute tricking the endpoint into thinking it has passed an assessment when in fact it has not and requires remediation. Because the IMC has no way to verify that the assessment results were actually created by an authentic IMV it is unable to detect the falsified attribute or message.

5.2.3 Attribute Modification

This attack could allow an active attacker capable of intercepting a message to modify an IF-M message attribute to a desired value to ease the compromise of an endpoint. Without the ability for message recipients to detect whether a received message contains the same content as what was originally sent, active attackers can stealthily modify the attribute exchange. For example, an attacker might wish to change the contents of firewall component's version string attribute to disguise the fact that the firewall is running an old vulnerable version. The attacker would change the version string sent by the firewall IMC to the current version number so the IMV's assessment passes while leaving the endpoint vulnerable to attack. Similarly

an attacker could achieve wide spread denial of service by altering large number of assessments' version string attribute to an old value so they repeatedly fail assessments even after a successful remediation. By sending a lower value the IMV continues to believe that the endpoint is running old, potentially vulnerable versions of the firewall that does not meet network compliance policy so therefore is not allowed to join the network.

5.2.4 Attribute Replay

Another potential attack against an unprotected IF-M message attribute exchange is to exploit the lack of a strong binding between the attributes sent during an assessment to the specific endpoint. Without a strong binding of the endpoint to the measurement information, an attacker could record the attributes sent during an assessment of a compliant endpoint and later replay those attributes so that a non-compliant endpoint can now gain access to the network or protected resource. This attack could be employed by a network MiTM that is able to eavesdrop and proxy message exchanges or using local rogue agents on the endpoints. Assessments lacking some form of freshness exchange could be subject to replay of prior assessment data even if it no longer reflects the current state of the endpoint.

5.2.5 Attribute Insertion

Similar to the attribute modification attacks, an adversary wishing to include one or more attributes or IF-M messages inside a valid assessment may be able to insert the attributes or messages without detection by the recipient. Even if authentication of the parties is present during an IF-M exchange, if no per-message and per-session integrity protection is present an attacker can add information to the assessment possibly causing incorrect assessment results. For example an attacker could add attributes to the front of an IF-M message to cause an assessment to succeed even for a non-compliant endpoint particularly if it knew that the recipient ignored repeated attributes within a message. Similarly if an IMC or IMV always generated an error if it saw unexpected attributes, the attacker could cause failures and denial of service by adding attributes or messages to an exchange.

5.2.6 Denial of Service

A variety of types of denial of service attacks are possible against the IF-M message exchange if left unprotected to untrusted parties along the communication path between the IMC and IMV. Normally the IF-T exchange is bi-directionally authenticated which helps to prevent MiTM on the network from active proxies but transparent message routing gateways may still exist on the communication path and can modify the integrity of the message exchange unless adequate integrity protection is provided. If the MiTM or other entities on the network can send messages to the TNCC or TNCS that appear to be part of an assessment these messages could confuse or cause the IMC and IMV to perform unnecessary work or take incorrect action. Several example denial of service situations are described in section 5.2.3 and 5.2.5. Many potential denial of service examples exist including flooding messages to IMC or IMV, sending very large messages containing many attributes, and repeatedly asking for resource intensive operations.

6 Privacy Considerations

The IF-M protocol is designed to allow for controlled disclosure of security relevant information about an endpoint specifically for the purpose of enabling an assessment of the endpoint's compliance with network policy. The purpose of this protocol is to provide visibility into the state of the protective mechanisms on the endpoint in order for the IMVs and TNCS to determine whether the endpoint is up to date and thus having the best chance of being resilient in the face of malware threats. One risk associated with providing visibility into the contents of an endpoint is the increased chance for exposure of privacy sensitive information without the consent of the user.

While this protocol does provide the IMV the ability to request specific information about the endpoint, the protocol is not open ended, bounding the IMV to only query specific information (attributes) about specific security features (component types) of the endpoint. Each IF-M message is explicitly about a single component from the list of components in section 3.4. These components include a list of security related aspects of the endpoint that affect the ability of the endpoint to resist attacks and thus are of interest during an assessment. Discretionary components used by the user to create or view content are not on the list as they are more likely to have access to privacy sensitive information. Similarly, IF-M messages contain a set of attributes which describe the particular component. Each attribute contains generic information (e.g. product information or versions) about the component so is unlikely to include any user specific or identifying information. This combination of limited set of security related components with non-user specific attributes greatly reduces the risk of exposure of privacy sensitive information. Vendors that choose to define additional component types and/or attributes within their name space are encouraged to provide similar constraints.

Even with the bounding of standard attribute information to specific components; it is possible that individuals might wish to share less information with different networks they wish to access. For example, a user may wish to share more information when connecting or being re-assessed by the user's employer network than made available to the local coffee shop wireless network. While these situations do not impact the protocol itself, they do suggest that IMC implementations should consider supporting a privacy filter allowing the user and/or system owner to restrict access to certain attributes based upon the target network. The underlying IF-T protocol authenticates the network's TNCS at the start of an assessment, so identity could be made available to the IMC so per-network privacy filtering is possible. Network owners should make available a list of the attributes they require to perform an assessment and any privacy policy they enforce when handling the data. Users wishing to use a more restricted privacy filter on the endpoint may risk not being able to pass an assessment and thus not gain access to the requested network or resource.

7 IF-M Message Flows

IF-M is a multi-roundtrip message exchange protocol between a registered set of IMCs and IMVs in order to perform an assessment of the security state of an endpoint. As discussed in the use cases in section 2.2, an assessment can be triggered by the TNCC or TNCS and the individual messages could each be filling one or more of the following purposes:

- Requesting measurements from IMCs,
- Providing measurements to IMVs (possibly leveraging earlier assessment results),
- Sending the results of an IMV's evaluation,
- Sending remediation instructions.

This section discusses three examples of how the message flow might proceed during an assessment. These examples are not intended to limit the complexity or ordering of the messages but rather to establish a general context for understanding how each message's attributes might relate to other attributes sent later in the handshake. As discussed in section 3.1, the IF-M message exchange occurs in a round-robin manner. Each party (IMC or IMV) sends as many attributes in as many messages as desired and then relinquishes control over the exchange to the peer(s) to reply. For simplicity, the examples will not vary the number of attributes included in each message.

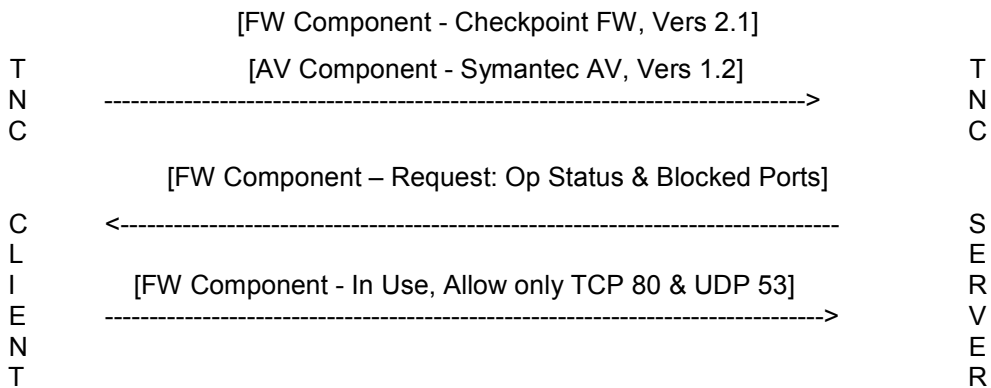
For the example flow diagrams in the following subsections, the following grammar is used to briefly summarize each IF-M message:

[<Component Name> - <Comma separated list of attributes>]

7.1 Simple IMC Initiated Example

One expected common usage model for the TNC architecture has a set of IMCs being notified by the TNCC that a network connection is being initiated to a particular network and asked to provide measurement information based upon local policy. In this simple example each assessed component has a single IMC and a corresponding IMV that is able to evaluate its compliance to network policy. For this example no IF-M security protocol is required.

Specifically, the TNCC has an anti-virus (AV) and firewall (FW) IMC and the TNCS has the corresponding AV and FW IMVs. The IMCs are pre-configured to send product information and version attributes upon connection to this network. The FW IMV will request additional information (operational status and which networking ports are blocked) before making a decision. Neither IMV decided to inform the IMC whether it passed the assessment.



Simple IMC Initiated Example IF-M Message Flows

C	[AV Component from IMC #1 - Norton AV, Vers 9, In Use]	S
L	[AV Component from IMC #9 - Trend AV, Version 1.1, Installed]	E
I	----->	R
E		V
N	[AV Component to only IMC #1 - AV Failed, AV Remed. Instructs]	E
T	<-----	R

Multiple IMCs for Single Component IF-M Message Flows

This time notice that both AV IMCs respond with measurement information for its associated AV component, one for Norton and the other for the Trend Micro product. Because each IMC is only reporting on a single product it doesn't need to include a correlation identifier. For example, the AV IMV can correlate which attributes apply to the Norton AntiVirus product by checking the source IMC of the message and remembering (if the IMV intends to send additional attribute queries about AV) that IMC #1 always reports on Norton AntiVirus. The only situation where a separate correlation identifier is needed is when a single IMC can report upon multiple different products associated with a single component type.

In this case the AV IMV decides the Norton AntiVirus product is out of date. The IMV does not want the Norton remediation instructions to be attempted by the Trend IMC so it needs a way to exclusively communicate with the Norton IMC. Recall that normally all IMCs who registered an interest for a component type will receive it, so we need a way to indicate in the message that only the Norton IMC should act upon the remediation instructions.

There are two different ways to request only the Norton IMC act upon the message. The first involves using the security protocol to encrypt the message using keys shared only between the Norton IMC and AV IMV. The other lighter weight approach involves the use of an exclusive destination capability in IF-TNCCS. For this example, the IMV tells IF-TNCCS that the IF-M message is exclusively for processing by IMC #1. The TNCC will only deliver the message to the Norton IMC for processing. These techniques could be used for any type of exclusive communication with each IMC. For instance the IMV could request different information from each of the IMCs in different IF-M messages by requesting IF-TNCCS mark each message for exclusive delivery. After IMC #1 performs the needed remediation the TNCC could request a reassessment to verify its compliance and re-gain network access.

8 References

8.1 Normative References

- [KEYWORDS] S. Bradner, "Keywords for use in RFCs to Indicate Requirement Levels", <http://www.ietf.org/rfc/rfc2119.txt>, IETF, March 1997.
- [RFC2279] F. Yergeau, "UTF-8, a transformation format of ISO 10646", <http://www.ietf.org/rfc/rfc2279.txt>, IETF, January 1998.
- [RFC3339] G. Klyne, C. Newman, "Date and Time on the Internet: Timestamps", <http://www.ietf.org/rfc/rfc3339.txt>, IETF, July 2002.
- [RFC3986] T. Berners-Lee, R. Fielding, L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", <http://www.ietf.org/rfc/rfc3986.txt>, IETF, January 2005.

8.2 Informative References

- [IF-ARCH] Trusted Computing Group, "TNC Architecture for Interoperability", https://www.trustedcomputinggroup.org/specs/TNC/TNC_Architecture_v1_2_r4.pdf, May 2007.
- [IF-M-SEC] Trusted Computing Group, "IF-M Security: Bindings to CMS", specification in progress, February 2008.
- [IF-T] Trusted Computing Group, "TNC-T: Protocol bindings for Tunneled EAP Methods", https://www.trustedcomputinggroup.org/specs/TNC/TNC_IFT_v1_1_r10.pdf, May 2007.
- [IF-TNCCS] Trusted Computing Group, "TNC IF-TNCCS", https://www.trustedcomputinggroup.org/specs/TNC/TNC_IF-TNCCS_v1_1_r15.pdf, October 2006.
- [IF-TNCCS-SOH] Trusted Computing Group, "TNC IF-TNCCS: Protocol Bindings for SoH",

https://www.trustedcomputinggroup.org/specs/TNC/IF-TNCCS-SOH_v1.0_r8.pdf, May 2007.

[IF-IMC] Trusted Computing Group, "TNC IF-IMC",
https://www.trustedcomputinggroup.org/specs/TNC/TNC_IFIMC_v1_2_r8.pdf, October 2006.

[IF-IMV] Trusted Computing Group, "TNC IF-IMV",
https://www.trustedcomputinggroup.org/specs/TNC/TNC_IFIMV_v1_2_r8.pdf, October 2006.