NG112 Testing

Interoperability, suitability and conformity

EUROPEAN EMERGENCY NUMBER ASSOCIATION

NG112 Testing

Interoperability, suitability and conformity

Version 1.0

Publication date 16/01/2024

Status of document: FINAL

Authors:

Michael Proestler, GridGears

EENA
European Emergency Number Association
EENA 112

Avenue de la Toison d'Or 79, Brussels, Belgium

T: +32/2.534.97.89

E-mail: info@eena.org

Legal Disclaimer

Table of Contents

1. Executive Summary

NG112 Testing: Interoperability, suitability and conformity

When tendering and purchasing a Next Generation 112 (NG112) system, the validation and assurance of standard conformity and interoperability of the related core services is essential. Standard conformity and interoperability provide the foundation for extensibility, interoperability and future innovation of emergency communications. Therefore, vendors' activities in conformance testing, interoperability testing and their participation in industry events, such as ETSI NG112 Plugtests or NENA Industry Collaboration Events (ICE) are good selection criteria during the tendering process.

Conformance testing, interoperability testing and participation in industry events are multiple complementary approaches for evaluation, each focusing on different aspects and providing unique value. Conclusions regarding standard conformity and interoperability can only be drawn when considering all of them together.

This document intends to…..

Inform readers on the most important considerations when tendering and purchasing a Next Generation 112 (NG112) system

This document contains…..

- An introduction to NG112 core services
- Conformance testing
- Interoperability testing
- Plugtests and collaboration events
- The purpose and goals, set up and execution, and benefits and downsides of the above

2. Table of Acronyms

NG	Next Generation
ETSI	European Telecommunications Standards Institute
NENA	National Emergency Number Association (USA)
ICE	Industry Collaboration Event
IP	Internet Protocol
BCF	Border Control Function
ESRP	Emergency Services Routing Proxy
ECRF	Emergency Call Routing Function
LIS	Location Information Service
TS	Technical Specification
SUT	System Under Test
IUT	Implementation Under Test
ESInet	Emergency Services IP Network
TP	Test Purpose
URI	Uniform Resource Identifier
SIP	Session Initiation Protocol
LoST	Location-to-Service Translation
HELD	HTTP-Enabled Location Delivery

3. Introduction

Accompanied by the transition to packet-switched networks, the NG112 architecture with its core services provides the foundation of modern IP-based emergency communications. As stakeholders currently engage in the crucial process of tendering for an NG112 deployment, an understanding of the different testing procedures becomes essential.

This document provides a comprehensive understanding of the various testing methods applicable to NG112 core services. From conformance testing, which validates adherence to ETSI standards, to interoperability testing, ensuring seamless integration and communication between different components and plugtests/industry events fostering a collaborative testing environment between multiple companies, each method is discussed in detail.

4. NG112 Core Services

4.1 What's in Scope?

Testing NG112 Core Services, such as Border Control Function (BCF), Emergency Service Routing Proxy (ESRP), Emergency Call Routing Function (ECRF) or Location Information Service (LIS) is always related to the specified interfaces and protocols in ETSI TS 103 479.

Test cases and scenarios are developed purely based on the standards and are independent of vendors' solutions. As a result, only features and functionalities covered by the standard are dealt with in the three testing approaches.

Non-functional requirements include redundancy mechanisms, performance criteria and configuration capabilities. Although they are a crucial aspect when it comes to mission critical systems in public safety, they are often heavily dependent on the actual use case, deployment options, hardware requirements and software architecture, which cannot be generalised. Therefore, those non-functional requirements are not standardised and consequently are out of scope for this document.

It is worth mentioning that the testing methods are in no way a replacement for the vendors' own quality assurance or the acceptance tests by the customer, which then might also include the non-functional requirements mentioned above.

4.2 Closed-Box Strategy

Considering the standardised interfaces, protocols, payloads and capabilities, those are the only ways of interacting with the NG Core Services while testing. This is often described as closed-box testing, where the functionality is assessed without any detailed knowledge of the internal architecture, code structure or logic. The focus is primarily on the inputs and outputs.

Figure 1: Closed-Box Approach

A test case or test scenario triggers an input and verifies the resulting output. Depending on the testing approach, this might be automated, semi-automated or even manual steps and/or interactions.

The system that represents the box is often referred to as the System Under Test (SUT) or Implementation Under Test (IUT) and can range from a single core service to a full-blown Emergency Services IP Network (ESInet) with multiple core services or even to multiple ESInets.

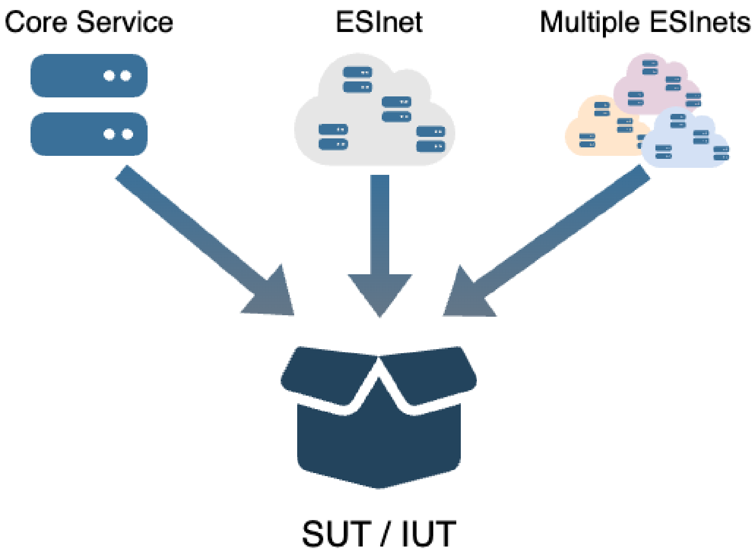

Figure 2: SUT/IUT Range

5. Conformance Testing

5.1 Purpose and Goals

Conformance testing verifies that a core service implements the standardised interfaces, protocols and capabilities and provides a foundation for later certification programs. Every single conformance test scenario is traceable to requirements in the standards and verifies a certain behaviour of the IUT, based on input and outputs.

TP Id	TP_LIS_HELD_BV_03
Test Objective	IUT successfully responds with a Reference when it receives a location request with location type locationURI and exact attribute
Reference	ETSI TS 103 479 [1], clause 6.5.1 EENA Next Generation 112 LTD [i.3], clause 4.10 IETF RFC 5985 [3]
PICS Selection	PICS_LIS_HELD_BY_REFERENCE_GET and PICS_LIS_GEOMETRY_CIRCLE

Initial Conditions

```
with {
                the IUT has a Mapping containing
                        uri             indicating value PX_URI,
                           location      indicating value PX_CIRCLE

            }
```

Expected Behaviour

```
ensure that {
                when {
                    the IUT receives an LocationRequest containing
                        uri         corresponding to PX_URI,
                        locationType containing
                            exact    indicating value True,
                            values    indicating value "locationURI"


                  }
                  then {
                    the IUT sends a LocationResponse containing
                        "locationUri"    carrying any URI

                  }
            }
```

Figure 3: ETSI TS 103 650-1 V1.2.1 Example of a Test Purpose / Test Scenario

The figure above illustrates the structure of a test purpose / test description, which provides the base for the later implementation of the actual test case. In this example from ETSI TS 104 650-1 V1.2.1, the initial conditions refer to the required configuration. In this case, the LIS requires a specific Uniform Resource Identifier (URI) to map to a certain circle location. The expected behavior refers to the automatically executed steps by the test framework. The when section refers to the input and the then section to the evaluated/expected output.

Conformance test cases are usually built and implemented on the foundation of a test framework in order to make them repeatable and consistently executable and with the least amount of manual interaction necessary.

Requirements and features are either optional or mandatory. Test cases are then selected based on a matrix of the vendor's capabilities and customer requirements. The figure below lists the requirements for the various ECRF features.

Table 4: ECRF features

Item	Name of field	Reference	Status	Support
Prerequisite: Table 1/3				
1	Does the IUT support Find Service	IETF RFC 5222 [5], clause 14	m	
2	Does the IUT support List Services	ETSI TS 103 479 [1], clause 6.4	m	
3	Does the IUT support List Services by Location			
4	Does the IUT support Point	IETF RFC 5222 [5], clause 12.2 IETF RFC 5491 [8], clause 5.2.1	m	
5	Does the IUT support Circle	IETF RFC 5222 [5], clause 12.2 IETF RFC 5491 [8], clause 5.2.3	m	
6	Does the IUT support Ellipse	IETF RFC 5222 [5], clause 12.2 IETF RFC 5491 [8], clause 5.2.3	o	
7	Does the IUT support Arcband	IETF RFC 5222 [5], clause 8.2 IETF RFC 5491 [8], clause 3.2	o	
8	Does the IUT support Polygon	IETF RFC 5222 [5], clause 12.2 IETF RFC 5491 [8], clause 5.2.3	o	
9	Does the IUT support Loop detection	IETF RFC 5222 [5], clause 6	m	
10	Does the IUT support Redirection	IETF RFC 5222 [5], clause 8.3.3	m	
11	Does the IUT support subscriptions to Element State	ETSI TS 103 479 [1], clause 5.3	o	
12	Does the IUT support subscriptions to Security Posture	ETSI TS 103 479 [1], clause 5.3	o	
13	Does the IUT support subscriptions to Service State	ETSI TS 103 479 [1], clause 5.3	o	
14	Does the IUT support TLS mutual authentication	ETSI TS 103 479 [1], clause 5.5	o	

Figure 4: ETSI TS 103 650-1 V1.2.1 - ECRF Features

5.4 Setup and Execution

Conformance test cases often require a base configuration that core services need to be provisioned with. This configuration might include network specific parameters such as downstream elements of an ESRP or mapping configurations for an ECRF. In general conformance test cases, it is advisable to try to limit manual interactions although sometimes this is unavoidable. For example, the configuration of the core service might need to change between test cases in order to verify a certain behaviour or a manual verification/interaction is necessary to execute the test case.

Generally speaking, the test framework "wraps" around the IUT, provides inputs and evaluates the outputs.

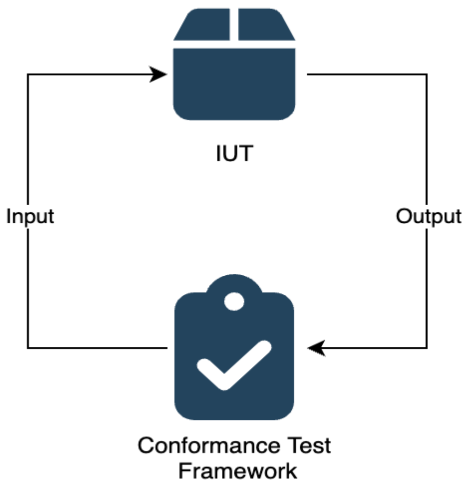

Figure 5: Conformance Test Framework Setup

Some basic examples for different core services and their input/outputs are as follows:

Core Service	Input	Output
Border Control Function	SIP Invite	SIP Invite
Emergency Service Routing Proxy	SIP Invite	SIP Invite
Emergency Call Routing Function	LoST Request	LoST Response
Location Information Service	HELD Request	HELD Response

In more advanced test cases, the test framework needs to simulate all connected core services in order to verify the standardized behavior of the IUT.

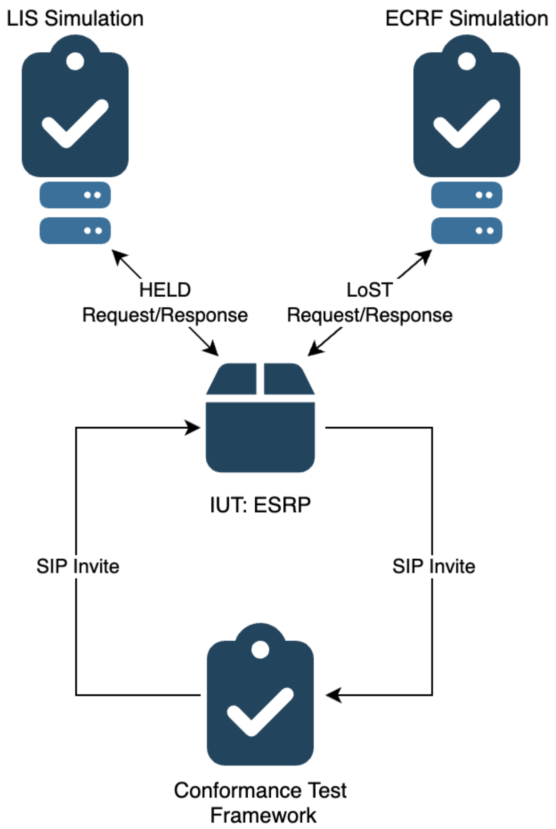

Figure 6: Example of Advanced Test Case Simulation

In the example above, the conformance test framework simulates a LIS and an ECRF in order to verify the following:

- Interaction between ESRP and LIS (HELD)
- Interaction between ESRP and ECRF (LoST)
- Interaction between ESRP and the downstream element (SIP)

5.3 Benefits and Downsides

Conformance testing objectively proves that interfaces, protocols and capabilities are supported and implemented according to the standard. The successful evaluation of a core service through conformance testing provides evidence that the specific core service adheres to the standards and implements the required behaviour regarding the covered functionality by the test cases. Since conformance test cases are mostly automated, they can also be integrated into a vendor's development process to ensure standard conformity throughout regression testing.

Furthermore, it is fair to say that if two interacting services each pass their specific conformance test cases, the chance of them being interoperable with each other increases, however it cannot be guaranteed. It is worth noting though that there might be multiple ways of implementing certain features and/or functionalities, as is often the case in complex standards and systems. This sometimes leads to different "variations" of conformance test cases depending on the vendor's implementation. This means vendors might provide conformant implementations, which are not interoperable with each other.

On the downside, conformance testing and the later establishment of a certification program are associated with high efforts and costs as test cases need to be developed and implemented. Test cases then also need to be evaluated themselves through different vendor implementations in order to rule out errors within the test cases themselves.

6. Interoperability Testing

6.1 Purpose and Goals

The main goal of interoperability testing is to verify that two or multiple systems/services can interact and communicate with each other in different scenarios. Within the NG112 architecture and its core services, interoperability is often the result of high standard conformity of the interacting services. However, since some features might be correctly implemented in different ways, interoperability testing provides another layer of assurance that the actual combination of those services works correctly.

In contrast to conformance testing, where the IUT is a single core service, interoperability testing requires multiple components. Furthermore, interoperability testing is not restricted to core services specified in the standards. As the NG112 architecture and its core services provide the foundation for IP-based emergency communications, other services and technologies (not specified by standards) might leverage those services and interfaces while providing additional value and functionalities.

6.2 Setup and Execution

The most basic scenarios for interoperability testing are protocols and interfaces based on a request/response mechanism, e.g. Location-to-Service Translation (LoST) or HTTP-Enabled Location Delivery (HELD). Based on a certain configuration, the client can send different requests and evaluate the responses. This is quite similar to conformance testing, but instead of a testing framework, the integrating service is triggering the request and verifies if it is able to understand and process the response.

In general, interoperability testing between two components/services can be automated but is dependent on the vendors' implementations.

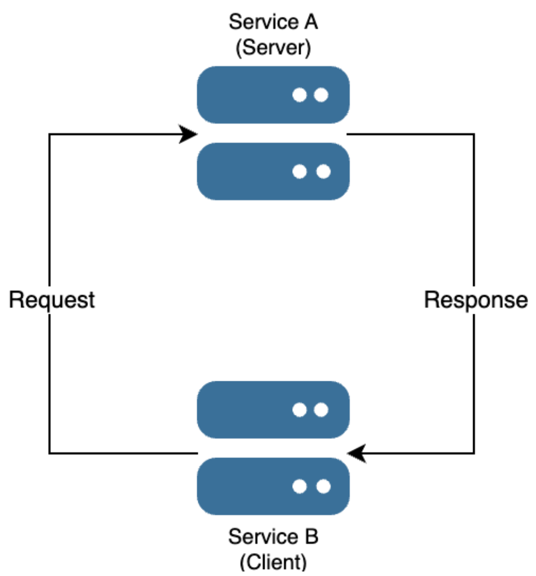

Figure 7: Basic Interoperability Setup

In a more advanced setup, multiple components and services can be combined. However, since the service triggering the input might not be the same as receiving the output, automation is far more complex, which is why running test cases in advanced setups usually requires semi-automated and manual interactions.

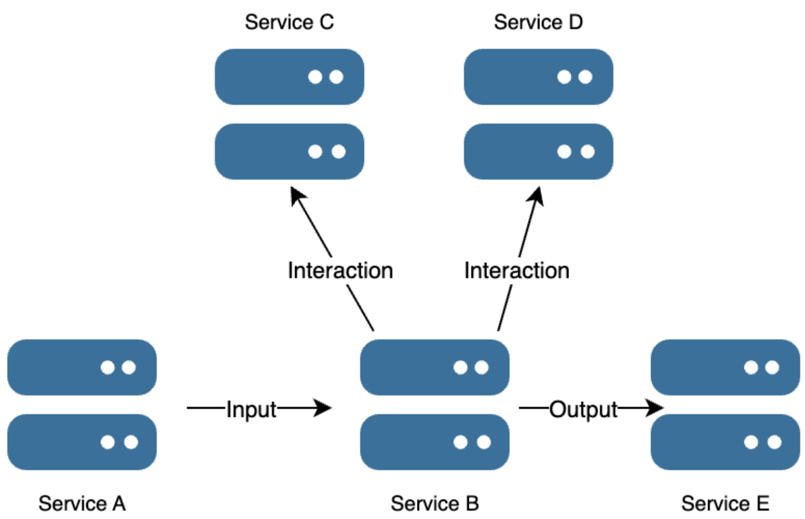

Figure 8: Advanced Interoperability Setup

6.3 Benefits and Downsides

Although multiple implementations can be interoperable with each other, this does not provide any proof of standard conformity in general. However, if those components were truly developed independently based on standards, successful interoperability between them indicates at least a certain level of common understanding of the interfaces and standard conformity.

One downside of interoperability testing, especially in more advanced setups, is to identify the source of interoperability issues. Usually requests and responses need to be manually evaluated and verified against the standards and, in an advanced setup, a test failure might be the result of multiple independent issues. In those cases, not only the initial request and final response, but also the message exchanges between intermediate components might have to be examined.

7. Plugtests & Collaboration Events

7.1 Purpose and Goals

Industry events such as ETSI Plugtests and Industry Collaboration Events (ICE) are quite similar to interoperability testing. Multiple vendors interconnect their implementations and run predefined scenarios usually through a full NG112 architecture. Having many different vendors of core services allows for the effective running of multiple permutations and combinations of different core services and the ability for the vendors to verify their understanding of the standards, contribute to the discussion and identify missing parts within the standard.

It is worth mentioning that during ETSI Plugtests, government bodies, policy makers and local authorities are also invited to attend as observers.

7.2 Setup and Execution

The setup usually consists of a full NG112 architecture with its core services. Participating vendors provide information upfront about which core services they provide and what features are implemented and supported. Based on this information from the vendors, multiple test scenarios are developed. Those scenarios are then executed using different permutations of the vendors' core services.

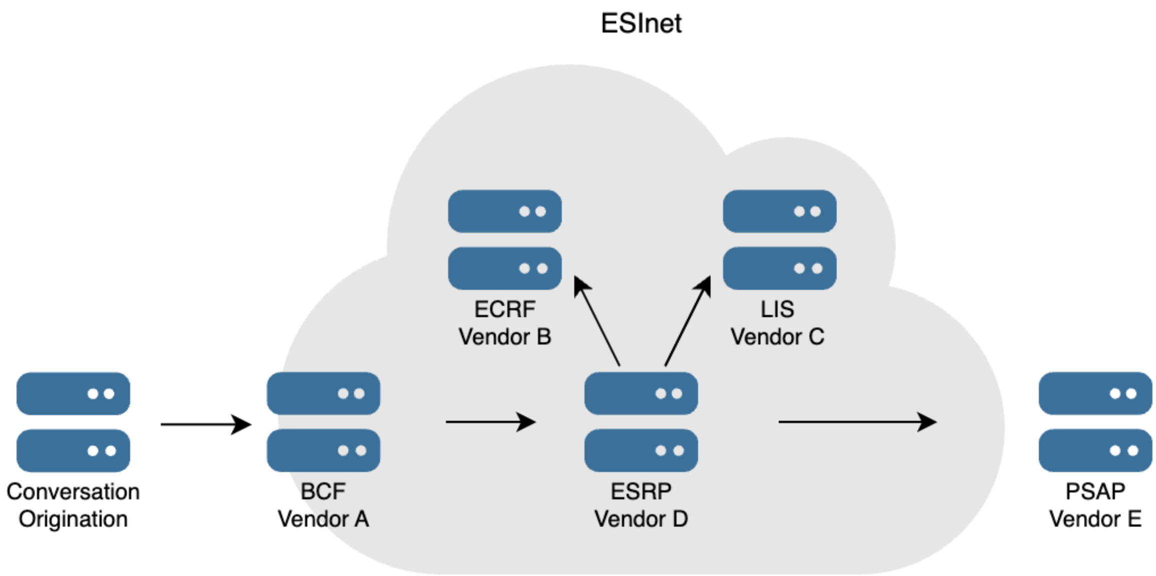

Figure 9: Full Setup

Similar to interoperability testing, most of the evaluation and validation is performed manually.

7.3 Benefits and Downsides

Benefits are similar to interoperability testing, but on a much larger scale, due to interaction with many different vendors of different core services. Participating companies additionally benefit from multiple perspectives of the standards and might increase the standard conformity of their products due to valuable interactions with other vendors.

On the downside, the identification of issues requires a lot of effort due to the complexity of the overall system. Additionally, a vendor's participation as such does not provide any information about the standard conformity of their products.

8. Summary

Each testing method offers distinct value in the validation and assurance of standard conformity and interoperability. Generally, these methods complement and influence each other. As an example, when two vendors pass a wide selection of conformance test cases, chances are high that they are also interoperable. Similarly, if one vendor's product is interoperable with products of many other vendors, chances are high that the product adheres to the standards. In general, conformance testing provides a solid foundation and paves the way for later interoperability. In addition, industry events such as ETSI NG112 Plugtests or Industry Collaboration Events (ICE) are a great way of interacting with many vendors and performing interoperability testing at scale.

Authorities and stakeholders tendering and purchasing a NG112 architecture should put emphasis on standard conformity by evaluating the vendors activities through conformance and interoperability testing as well through participation in industry events in order to create a solid foundation for future innovations of their emergency communications solutions.