**3GPP TSG- Meeting # *-r1***

**, , -**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.2* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  |  | **CR** |  | **rev** |  | **Current version:** |  |  |
|  | | | | | | | | |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* | | | | | | | | |
|  | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME |  | Radio Access Network |  | Core Network |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | Minimized kernel network functions (TC\_ IP\_MULTICAST\_HANDLING) | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** |  | | | | | | | | | |
| ***Source to TSG:*** | S3 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | SCAS\_5G\_Ph2 | | | | |  | ***Date:*** | | |  |
|  |  | | | |  | |  | | |  |
| ***Category:*** | F |  | | | | | ***Release:*** | | |  |
|  | *Use one of the following categories:* ***F*** *(correction)* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | | | | | | | *Use one of the following releases: Rel-8 (Release 8) Rel-9 (Release 9) Rel-10 (Release 10) Rel-11 (Release 11) … Rel-16 (Release 16) Rel-17 (Release 17) Rel-18 (Release 18) Rel-19 (Release 19)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | Rewrite of test case so it is objective and meets the NESAS purpose. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Rewriting of execution steps so it is objective and meets the stated purpose. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | Text could be interpreted differently by different Test Labs, making it extremely difficult to be certain that all tests are being conducted such that the results would be the same, regardless of which Test Lab did the work. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 4.3.3.1.2 Minimized kernel network functions | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  | **X** | Other core specifications | | | | TS/TR ... CR ... | | |
| ***affected:*** | |  | **X** | Test specifications | | | | TS/TR ... CR ... | | |
| ***(show related CRs)*** | |  | **X** | O&M Specifications | | | | TS/TR ... CR ... | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

**\*\*\* BEGIN OF CHANGE \*\*\***

##### 4.3.3.1.2 Minimized kernel network functions

*Requirement Name*: Minimized kernel network functions.

*Requirement Reference*: In accordance with industry best practice

*Requirement Description*:

Kernel based network functions not needed for the operation of the network element shall be deactivated.

In particular the following ones shall be disabled by default:

Note: Void

- Proxy ARP (to prevent resource exhaustion attack and man-in-the-middle attacks.

- Directed broadcast (to prevent Smurf, Denial of Service attack and others like it*.*

- IPv4 Multicast handling. In particular all packets with IP source or destination address belonging to the multicast IP ranges (224.0.0.0 through 239.255.255.255) shall be discarded by default and multicast route caching and forwarding shall be disabled to prevent smurf and fraggle attacks. A configuration option shall be available to enable the IPv4 multicast handling if required.

- Gratuitous ARP messages (to prevent ARP Cache Poisoning attacks [ef]). A Gratuitous ARP request can be used mainly to inform the neighbours about the change in the MAC for the specified IP and consequently to update their ARP tables or to update the switches with the new MAC address or to defend link-local IP addresses in the Zeroconf protocol. By default, the network product shall not send Unsolicited ARP and any incoming Gratuitous ARP requests shall be discarded.

Note: The above text does not preclude that Gratuitous ARP can be enabled in certain deployment scenarios.

Answering routine for broadcast ICMP packets. In particular all ICMP ECHO and TIMESTAMP requests sent to network product via broadcast/multicast shall not be answered by default.

*Threat References*: TR 33.926 [4]

*Test Case*:

**Test Name**: TC\_PROXY\_ARP\_DISABLING

**Purpose:**

Verify that the Proxy ARP feature is disabled by default on the network product. In particular this test case verifies that the network product does not respond to ARP requests intended for another host.

**Procedure and execution steps:**

**Pre-Conditions:**

- The network product shall have at least 2 different physical or logical Ethernet interface IF1 and IF2. E.g.

- Host 1 is connected to IF1 on subnet A (for example 172.16.10.0/16).

- Host 2 is connected to IF2 on subnet B (for example 172.16.20.0/24).

- Network traffic analyser on the network product (e.g. TCPDUMP) or an external traffic analyser directly connected to the network product is available.

**Execution Steps**

1. If the feature is available in a configuration file, verify that it is disabled by default.

2. Broadcast an ARP request from Host 1 on Subnet A to discover the MAC of Host 2 on subnet B. Since the ARP request is a broadcast, it reaches all nodes in the Subnet A, which include the IF1 interface of the network product, but it does not reach Host 2.

3. Verify that the network product correctly receives this packet but that it does not send an ARP reply to Host 1 with its own MAC address.

**Expected Results:**

No Arp Reply is received by Host 1.

**Expected format of evidence:**

Pcap trace, snapshot of ARP Cache of Host 1

**Test Name**: TC\_DIRECTED\_BROAD\_DISABLING

**Purpose:**

Verify that the Directed broadcast is disabled by default on the network product. In particular this test case verifies that a packet received by a network product whose destination address is a valid broadcast address is dropped.

**Procedure and execution steps:**

**Pre-Conditions:**

- The network product has at least 2 different physical or logical Ethernet interface IF1 and IF2.

- Host 1 is connected to IF1 on Subnet A and Host 2 is connected to IF2 on Subnet B.

- Network traffic analyser on the network product (e.g. TCPDUMP) or an external traffic analyser directly connected to the network product is available.

**Execution Steps**

1. If the feature is available in a configuration file, verify that it is disabled by default.

2. Send an IP packet from Host 1 whose IP destination address is a valid broadcast address belonging to the subnet B.

3. Verify that the Host 2 on Subnet B does not receive the packet because it will be dropped by the network product, rather than being broadcasted.

**Expected Results:**

The packet is not broadcasted by the network product and Host 2 cannot receive it.

**Expected format of evidence:**

Pcap trace showing that packet from host 1only incomes to the network product.

**Test Name:** TC\_ IP\_MULTICAST\_HANDLING

**Purpose:**

Verify that IP Multicast is disabled by default on the network product. In particular this test case verifies that packets with IP source or destination address belonging to the multicast IP ranges (224.0.0.0 through 239.255.255.255) are not handled by the network product.

**Procedure and execution steps:**

**Pre-Conditions:**

- Network traffic analyser on the network product or an external traffic analyser directly connected to the network product is available.

- Network product

Capability:

NOT applicable in certain deployment scenarios where multicast needs to be enabled.

**Execution Steps**

1. Start a capture tool on the network product.

2. Send a multicast packet from a source IP address that belongs to the multicast IP range (224.0.0.0 through 239.255.255.255).

3. Capture the multicast packet on the network product.

4. Stop the capture tool on the network product.

5. Analyze the captured packet to verify that it was not forwarded by the network product.

**Expected Results:**

The multicast packet should not be forwarded by the network product.   
The test case is successful if the multicast packet is not forwarded by the network product.

**Expected format of evidence:**

Screenshot containing command output.

**\*\*\* END OF CHANGE \*\*\***