**3GPP TSG-SA3 Meeting #123 S3-252698-r2**

Goteborg, Sweden, 25 – 29 August 2025

|  |
| --- |
| *CR-Form-v12.1* |
| **CHANGE REQUEST** |
|  |
|  | **33.117** | **CR** | **0220** | **rev** | **-** | **Current version:** | **19.2.0** |  |
|  |
| *For* ***HE******LP*** *on using this form: comprehensive instructions can be found at http://www.3gpp.org/Change-Requests.* |
|  |

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

|  |
| --- |
|  |
| ***Title:***  | Adding NOTE to web server tests related to configuration files |
|  |  |
| ***Source to WG:*** | BSI (DE) |
| ***Source to TSG:*** | S3 |
|  |  |
| ***Work item code:*** | SCAS\_5GA |  | ***Date:*** | 2025-08-11 |
|  |  |  |  |  |
| ***Category:*** | F |  | ***Release:*** | Rel-20 |
|  | *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 canbe found in 3GPP TR 21.900. | *Use one of the following releases:Rel-8 (Release 8)Rel-9 (Release 9)Rel-10 (Release 10)Rel-11 (Release 11)…Rel-15 (Release 15)Rel-16 (Release 16)Rel-17 (Release 17)Rel-18 (Release 18)* |
|  |  |
| ***Reason for change:*** | Multiple test steps for web server test cases relate to web server configuration files to be checked.A web server may not always be a standalone application but could be directly integrated into the network product, missing a web server configuration file and use hardcoded web server settings. |
|  |  |
| ***Summary of change:*** | Added a NOTE to several test cases to notify the tester to omit test steps in such cases. |
|  |  |
| ***Consequences if not approved:*** | Test cases may fail due to the lack of web server configuration files regardless of the web server following the security requirements. |
|  |  |
| ***Clauses affected:*** | 4.3.4.2, 4.3.4.4, 4.3.4.5, 4.3.4.6, 4.3.4.7, 4.3.4.10, 4.3.4.12, 4.3.4.14 |
|  |  |
|  | **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:*** |  |

#### 4.3.4.2 No system privileges for web server

*Requirement Name*: No system privileges for web server.

*Requirement Reference*: In accordance with industry best practice

*Requirement Description*:

No web server processes shall run with system privileges. This is best achieved if the web server runs under an account that has minimum privileges. If a process is started by a user with system privileges, execution shall be transferred to a different user without system privileges after the start.

*Threat References*: TR 33.926 [4], clause 5.3.8, Elevation of privilege

*Test Case*:

***Test Name*:** TC\_NO\_SYSTEM\_PRIVILEGES\_WEB\_SERVER

**Purpose:**

Verify that the Web server is not run under system privileges.

**Procedure and execution steps:**

**Pre-Conditions:**

- The tester has needed administrative privileges.

- A tester machine is available.

- Recommended: an automatic assessment tool has been configured /script adapted in line with the Requirement Description.

**Execution Steps**

1. The tester checks that no web server processes run with system privileges. The tester checks that this is the case even for processes that may have been started by a user with system privileges.

a. The tester starts the web server process as web server user and checks process privileges.

b. If possible, the tester starts the web server process with system privileges and check if process privileges get dropped.

2. The tester checks in relevant system settings and web server configurations that a web server user is configured with minimal privileges needed to run the web server and the web server is executable by that user.

NOTE: The web server could not be a standalone application but could be integrated directly into (parts of) the network product and could therefore not use configuration files. Configuration could be done via command line parameters or simply be hardcoded into the application. In such cases the tester can omit test steps or parts of test steps related to web server configuration files.

**Expected Results:**

- There are no findings of web server processes that run with system privileges.

- System settings are set to ensure that no processes will run with system privileges.

**Expected format of evidence:**

- Log files / command line output and screen shots of test executions

- Part of web server and/or system configuration (plain text or screenshot) showing the configured user for the web server process

#### \*\*\*\* NEXT CHANGE\*\*\*\*

#### 4.3.4.4 No unused add-ons

*Requirement Name*: No unused add-ons

*Requirement Reference*: In accordance with industry best practice

*Requirement Description*: All optional add-ons and components of the web server shall be deactivated if they are not required. In particular, CGI or other scripting components, Server Side Includes (SSI), and WebDAV shall be deactivated if they are not required.

*Threat References*: TR 33.926 [4], clause 5.3.6.11, Unnecessary Services

*Test Case*:

***Test Name*:** TC\_NO\_UNUSED\_ADD-ONS

**Purpose:**

To verify that the Web server has deactivated unneeded add-ons and unneeded scripting components.

**Procedure and execution steps**

**Pre-Conditions:**

- The vendor has supplied a list of add-ons or scripting tools for Web server components needed for system operation, and that therefore need to be exempted from the test investigation.

- The tester has administrative privileges.

- A tester machine is available.

- Recommended: an automatic assessment tool has been configured / script adapted in line with the Requirement Description.

**Execution Steps**

1. Check that the web server is only running and listening on known ports (e.g. tcp port 80 and/or 443). Check that CGI or other scripting components, Server Side Includes (SSI), and WebDAV are deactivated if they are not required. See also guidance under 4.3.4.12.

2. Check that nothing else has been installed than the web server.

3. Check that relevant system settings and configurations are correct to ensure fulfilment of the requirement.

NOTE: The web server could not be a standalone application but could be integrated directly into (parts of) the network product and could therefore not use configuration files. Configuration could be done via command line parameters or simply be hardcoded into the application. In such cases the tester can omit test steps or parts of test steps related to web server configuration files.

**Expected Results:**

System settings and configurations have been found, for all Web components of the system, to ensure that all unneeded add-ons or script components are deactivated.

**Expected format of evidence:**

Log files and screen shots of test executions.

#### 4.3.4.5 No compiler, interpreter, or shell via CGI or other server-side scripting

*Requirement Name*: No compiler, interpreter, or shell via CGI or other server-side scripting.

*Requirement Reference*: In accordance with industry best practice

*Requirement Description*: If CGI (Common Gateway Interface) or other scripting technology is used, the CGI directory - or other corresponding scripting directory - shall not include compilers or interpreters (e.g. PERL® interpreter, PHP interpreter/compiler, Tcl interpreter/compiler or operating system shells).

*Threat Reference*: TR 33.926 [4], clause 5.3.6, Information disclosure

*Test Case*:

***Test Name*:** TC\_NO\_COMPILER\_FOR\_CGI

**Purpose:**

To verify that there are no compilers, interpreters or shell accessible via CGI or other scripting components.

**Procedure and execution steps**

**Pre-Conditions:**

- The tester has administrative privileges

- A tester machine is available.

- Recommended: an automatic assessment tool has been configured /script adapted in line with the Requirement Description.

**Execution Steps**

1. Consult the web server configuration to identify all directories used for CGI or other scripting components.

NOTE: The web server could not be a standalone application but could be integrated directly into (parts of) the network product and could therefore not use configuration files. Configuration could be done via command line parameters or simply be hardcoded into the application. In such cases the tester can omit test steps or parts of test steps related to web server configuration files.

2. Check that there are no compilers or interpreters (e.g., PERL® interpreter, PHP interpreter/compiler, Tcl interpreter/compiler or operating system shells) in the directory/directories used for CGI or for other scripting tools (including PERL®, PHP, and others).

**Expected Results:**

There are no compilers, interpreters or shells in directories accessible via CGI or other scripting components.

**Expected format of evidence:**

- Log files and screen shots of test executions.

- Part of web server configuration (plaintext or screenshot) showing all directories accessible by the CGI/scripting components.

- List of files (with types and permissions, if available) inside the directories accessible by the CGI/scripting components.

#### 4.3.4.6 No CGI or other scripting for uploads

*Requirement Name*: No CGI or other scripting for uploads.

*Requirement Reference*: In accordance with industry best practice

*Requirement Description*: If CGI or other scripting technology is used, all directories where the web server has write permissions shall be distinct from all directories containing CGI/script or executable code.

*Threat References*: TR 33.926 [4], clause 5.3.8.3, Folder Write Permission Abuse

*Test Case*:

**Test Name:** TC\_NO\_CGI\_OR\_SCRIPTING\_FOR\_UPLOADS

**Purpose:**

To ensure that directories with write permissions for the web server do not contain executable code such as CGI scripts.

**Procedure and execution steps:**

**Pre-Condition:**

If the web server is configured with CGI/Scripting on, this test applies.

**Execution Steps**

1. The tester identifies directories where the web server user has write permissions.

2. The tester verifies that these writable directories do not contain any executable scripts, CGI programs, or other executable code.

3. The tester verifies that directories configured for CGI/Scripting do not have write permissions for the web server.

NOTE: The web server could not be a standalone application but could be integrated directly into (parts of) the network product and could therefore not use configuration files. Configuration could be done via command line parameters or simply be hardcoded into the application. In such cases the tester can omit test steps or parts of test steps related to web server configuration files.

**Expected Results:**

Web server user writable directories are different from those containing executable code or the ones configured to be used for CGI/scripting.

**Expected format of evidence:**

A part of the configuration file / screenshot of the configuration showing that the web server is properly configured and the corresponding file system permissions.

#### 4.3.4.7 No execution of system commands with SSI

*Requirement Name*: No execution of system commands with SSI.

*Requirement Reference*: In accordance with industry best practice

*Requirement Description*: If Server Side Includes (SSI) is active, the execution of system commands shall be deactivated.

*Threat Reference*: TR 33.926 [4], clause 5.3.8, Elevation of privilege

*Test Case*:

**Test Name**: TC\_NO\_EXECUTION\_OF\_SYSTEM\_COMMANDS

**Purpose:**

To test whether it is possible to use the exec directive and if so, whether it can be used for system commands.

**Procedure and execution steps:**

**Pre-Condition:**

If the web server is configured with SSI active, this test applies.

**Execution Steps**

1. The tester checks whether execution of system commands is disabled in the web server configuration.

NOTE: The web server could not be a standalone application but could be integrated directly into (parts of) the network product and could therefore not use configuration files. Configuration could be done via command line parameters or simply be hardcoded into the application. In such cases the tester can omit test steps or parts of test steps related to web server configuration files.

2. The tester actually attempts to use the exec directive in an SSI file with and without system commands.

**Expected Results:**

- The execution of system commands via SSIs exec directive is disabled in the web server configuration.

- It is impossible to execute system commands via SSIs exec directive.

**Expected format of evidence:**

- A part of the configuration file / screenshot of the configuration showing that the web server is properly configured. For example, a configuration file that shows that the IncludesNOEXEC (Apache HTTP Server®) or ssiExecDisable (Microsoft® IIS) is set.

- Web server log while executing step 2.

#### \*\*\*\* NEXT CHANGE\*\*\*\*

#### 4.3.4.10 No directory listings

*Requirement Name*: No directory listings / Directory Browsing.

*Requirement Reference*: In accordance with industry best practice

*Requirement Description*: Directory listings (indexing) / "Directory browsing" shall be deactivated.

*Threat References*: TR 33.926 [4], clause 5.3.6.9, File/Directory Read Permissions Misuse

*Test Case*:

***Test Name*:** TC\_NO\_DIRECTORY\_LISTINGS

**Purpose:**

To verify that Directory listings / Directory browsing has been deactivated in all Web server components.

**Procedure and execution steps**

**Pre-Conditions:**

- The tester has administrative privileges

- A tester machine is available.

- The tester should have configured a script, or an automatic assessment tool adapted in line with the Requirement Description..

**Execution Steps**

1. The tester checks the web server configuration for Directory listings (indexing) / "Directory browsing" to be deactivated in all Web server components.

NOTE 1: The web server could not be a standalone application but could be integrated directly into (parts of) the network product and could therefore not use configuration files. Configuration could be done via command line parameters or simply be hardcoded into the application. In such cases the tester can omit test steps or parts of test steps related to web server configuration files.

2. The tester attempts directory listings on all endpoints (domains, subdomains and directories) offered by the web server.

NOTE 2: Whether directory listings have been deactivated could be done by checking the web server configuration file specifically the parameters related to directory listing. The directory listing could be turned off in the web server configuration file, and there is no activation capability.

NOTE 3: Directory listings could be obtained by entering a valid URL (e.g., /var/www/test\_1) that does not contain any index file.

**Expected Results:**

- Directory listing / Directory browsing has been deactivated in all Web server components configurations.

- The tester is unable to perform Directory listing / Directory browsing on all endpoints (domains, subdomains and directories) offered by the web server.

**Expected format of evidence:**

- Log files and screen shots of test executions

- Text excerpt of the web server configuration showing that directory listing is disabled

#### \*\*\*\* NEXT CHANGE\*\*\*\*

#### 4.3.4.12 Web server information in error pages

*Requirement Name*: Web server information in error pages.

*Requirement Reference*: In accordance with industry best practice

*Requirement Description*: User-defined error pages shall not include version information about the web server and the modules/add-ons used. Error messages shall not include internal information such as internal server names, error codes, etc. Default error pages of the web server shall be replaced by error pages defined by the vendor.

*Threat References*: TR 33.926 [4], 5.3.6.5, System Fingerprinting

*Test Case*:

***Test Name*:** TC\_NO\_WEB\_SERVER\_ERROR\_PAGES\_INFORMATION

**Purpose:**

To verify that error pages and error messages do not include information about the web server.

**Procedure and execution steps**

**Pre-Conditions:**

- The tester has needed administrative privileges.

- A tester machine is available.

- The vendor provides documentation on user-defined error pages (e.g. location, content, where configured) and messages.

- The vendor provides a list of potential parameters/commands to trigger events resulting in an http status code 3xx, 4xx, 5xx.

- The tester should have configured a script, or an automatic assessment tool adapted in line with the Requirement Description.

**Execution Steps**

1. The tester verifies that the web server configuration does replace default error pages with error pages defined by the vendor.

NOTE 1: The web server could not be a standalone application but could be integrated directly into (parts of) the network product and could therefore not use configuration files. Configuration could be done via command line parameters or simply be hardcoded into the application. In such cases the tester can omit test steps or parts of test steps related to web server configuration files.

2. The tester verifies that the vendor defined error pages do not contain information about the web server.

3. The tester triggers and captures at least one occurrence of the following HTTP status code classes:

a) Redirection error response (300-399)

b) Client error response (400-499)

c) Server error response (500-599)

NOTE 2: Possible error pages that could be displayed are: 3xx: redirection, 4xx: client errors, 5xx: server errors.

NOTE 3: The 3xx error pages could be triggered by permanent or temporary move of content to other URL and the page is found because redirected.

NOTE 4: The 4xx error page could be triggered by trying to access a URL pointing to a non-existent or restricted resource.

NOTE 5: The 5xx error page could be triggered by requesting a HTTP method the web server does not support or disabled (e.g. CONNECT, PUT, PATCH).

**Expected Results:**

Generated error pages and error messages do not include information about the web server.

**Expected format of evidence:**

Log files and screen shots of test executions

#### \*\*\*\* NEXT CHANGE\*\*\*\*

#### 4.3.4.14 Restricted file access

*Requirement Name*: Restricted file access.

*Requirement Reference*: In accordance with industry best practice

*Requirement Description*: Restrictive access rights shall be assigned to all files which are directly or indirectly (e.g. via links or in virtual directories) in the web server's document directory. In particular, the web server shall not be able to access files which are not meant to be delivered.

*Threat References*: TR 33.926 [4], clause 5.3.6.9, File/Directory Read Permissions Misuse

*Test Case*:

**Test Name:** TC\_RESTRICTED\_FILE\_ACCESS

**Purpose:**

To test whether the restrictive access rights are assigned to all files which are directly or indirectly in the web server's document directory and to verify whether path traversal is made improbable.

**Procedure and execution steps:**

**Pre-Condition:**

The web server is configured according to the manual

**Execution Steps**

1. The tester verifies that access rights on the servable content (meaning directories and files) is set to the following:

a. The files are owned by the user that runs the web server;

b. The files are not writable to others, except the web server's account;

2. The tester verifies that the user running the web server is an unprivileged account;

3. For Operating Systems that have chrooted environments, the tester verifies that the web server runs inside a jail or chrooted environment. If the chrooted environment is not used, the web server or system functionality can be used to restrict access to file directories.

NOTE: The web server could not be a standalone application but could be integrated directly into (parts of) the network product and could therefore not use configuration files. Configuration could be done via command line parameters or simply be hardcoded into the application. In such cases the tester can omit test steps or parts of test steps related to web server configuration files.

**Expected Results:**

- Name of user running the web server with the privileges of the account;

- Access rights of files and directories that the web server serves;

- Configuration that shows that the web server is in a chrooted environment, or restricted by accessing to file directories.

**Expected format of evidence:**

A part of the configuration file / screenshot of the configuration showing that the web server, the file access rights and the account running the web server is properly configured.