Audit Process

Introduction

An audit is a process of sequential transferring of audit packages from an E-SDC to the tax authority's system and handling the response generated by the system for the specific device.

There are two specific scenarios: **Remote Audit** and **Local Audit**.

NOTE:
Basic rules and processes described in this section apply to both scenarios. Details are explained in separate sections - see Remote Audit and Local Audit.

Depending on the scenario, an audit may be triggered periodically, if one or more invoices are created, or after the insertion of an external memory device into an E-SDC.

An audit is always an asynchronous process. Depending on the amount of data and means of communication, it can take from less than a second to a couple of hours.

Once the E-SDC receives a response from the Secure Element (signed invoice), it must be encrypted and stored in E-SDC's non-volatile memory.

An E-SDC device must be fully functional during an audit. The POS must be able to sign new invoices as long as the Secure Element permits. There must be a mechanism in place that is responsible for the continuous operation of the Secure Element and E-SDC while audit packages are being transmitted to the tax authority's system or an external memory unit.

Remote Audit

Remote audit is the process of transferring data to the tax authority's system using an internet connection. It is the most common way to perform audits for any device with a stable internet connection.

An E-SDC checks if TaxCore.API is reachable. If TaxCore.API is reachable, the E-SDC authenticates the tax authority's system by using a server-side certificate installed on the TaxCore.API endpoint, enabling HTTPS protocol. The tax authority's system authenticates the E-SDC using a digital certificate issued on the Secure Element and issues a token for that session.

The E-SDC starts sending audit packages, performing a series of audits until all the data stored on its non-volatile memory is audited.

A Remote audit is not the only audit option for E-SDC. If the network connection is not available due to the

interruption of the service or a missing GPRS modem or network card, E-SDC is able to perform a Local Audit.

Remote audit step-by-step:

1. E-SDC submits an audit package by invoking TaxCore.API service Submit Audit Package
2. Tax authority's system verifies the data and returns a response containing:
    - audit package status
    - a set of Commands (including the Proof of Audit command)

Local Audit

Local audit initiated by a taxpayer is a common scenario for devices that lack the ability to connect to the internet due to the technical limitations of the devices or limited infrastructure.

Unlike in Remote Audit process, during the Local Audit, the E-SDC doesn't submit the ARP file and audit packages to TaxCore.API. Instead, those files are copied to an SD Card or a USB Flash Drive.

Local audit step-by-step:

1. An audit is initiated by inserting an SD card or a USB Flash drive into an E-SDC device.
2. E-SDC signals the beginning of the audit to the Secure Element (invokes *Start Audit* APDU command);
3. E-SDC copies audit packages to external memory unit (e.g. SD card, USB flash drive), starting with the oldest unaudited package, in a piecemeal fashion, as described in E-SDC Stores Audit Files on SD Card or USB Drive.

NOTE:
Audit packages remain stored in E-SDC's local memory until it receives a Proof-of-Audit command from TaxCore.API

4. The Secure Element returns ARP (260 bytes) to the E-SDC;

NOTE:
When performing Local Audit, the ARP file should be generated and saved each time when at least one audit package is submitted to the tax authority’s system, by using an external memory unit (see File-based communication).

5.

An operator uploads the audit packages and the ARP file to the Tax Authority's system (by using the Taxpayer Administration Portal or the tax authority's Back Office Portal).

6. After the successful upload, a set of Commands (including the Proof of Audit command) are generated by the tax authority's system.
7. An operator downloads the commands to an external memory unit (by using the Taxpayer Administration Portal or the tax authority's Back Office Portal).
8. An operator inserts the external memory unit with the commands file into an E-SDC which processes the commands as described in E-SDC Executes Commands, i.e. passes the payload to the *End Audit* APDU command;

NOTE:
The Proof of Audit command might be missing due to many reasons, such as: not all packages have been submitted, the tax authority system is experiencing a delay in data processing... Perhaps, the Proof of Audit command for a specific local audit might never be returned.

9. The Secure Element resets its current unaudited amount to 0.00 and returns the OK message to the E-SDC;
10. The E-SDC generates the `Commands Execution Results` structure (as described in Commands) and saves it to a file on the external media, as described in E-SDC stores a command execution result to the SD card or USB drive.
11. The operator uploads the `Commands Execution Results` file to the Tax Authority's system (by using the Taxpayer Administration Portal or the tax authority's Back Office Portal).

Proof of Audit

Introduction

Proof-of-Audit (POA) is generated by the tax authority’s system once all expected audit packages have been received and securely stored on the tax authority’s system.

Even if, due to the failure of the EFD component or some other reason, the taxpayer is unable to send one or more audit packages, the Tax Authority still has the option to issue a Proof-of-Audit if it can determine the tax liability for that secure element.

A POA cycle begins with E-SDC initiating the *Start Audit APDU command* (**Audit Start**). A POA cycle finishes with the Tax Authority's system receiving a confirmation that the secure element has successfully executed the issued Proof-of-Audit command (**Audit End**).

NOTE:
This article describes the default operation mode of the *Proof of Audit Service*. However, upon the Tax Authority's decision, the service can use different strategies for issuing a proof-of-audit. For the currently applicable non-standard issuance strategies, see Currently Applicable Non-Standard Strategy for Issuing Proof of Audit.

There are two scenarios for obtaining a Proof-of-Audit:

- **As part of the Local Audit process**
- **Via direct communication with TaxCore.API as described below**

    1. E-SDC signals the beginning of the audit to the Secure Element (invokes *Start Audit* APDU command);
    2. The Secure Element returns ARP (260 bytes) to the E-SDC;
    3. E-SDC starts the audit by sending audit data (over HTTPS). ARP is submitted to the tax authority's system using the same communication channel;
    4. If the request is correct, the system returns the HTTP status code 200 (OK);
    5. Tax authority's system generates a set of Commands (including the Proof of Audit command);
    6. E-SDC submits a commands request to the tax authority's system via one of the following channels:
        - ♠ by invoking TaxCore.API service Notify Online Status
        - ♠ by invoking TaxCore.API service Submit Audit Package
    7. Tax authority's system returns the commands to the E-SDC as a response;
    8. E-SDC processes the commands as described in E-SDC Executes Commands, i.e. passes the payload to the *End Audit* APDU command;
    9. The Secure Element resets its current unaudited amount to 0.00 and returns an OK message to the E-SDC;
    10. E-SDC reports the results of the commands' execution to TaxCore.API as described in Notify Commands Processed

NOTE:
Regardless of which scenario was used to initiate the POA cycle, once the commands (containing the Proof-of-Audit command) have been generated by the tax authority's system, the cycle can be completed by using the other scenario.

POA cycle frequency

Audit Start should be initiated periodically, and the length of the period between two Audit Starts should be set dynamically. In regular conditions, there is no reason to initiate an Audit more often than every 30 minutes; although this can be extended to a period of a couple of hours as well (depending on the turnover). There is no need to initiate unnecessary Audits if the secure element(s) does not register sale amounts that will cause it to cross its assigned limit.

However, if the secure element is reaching its limit, the Audit should be initiated immediately and the period

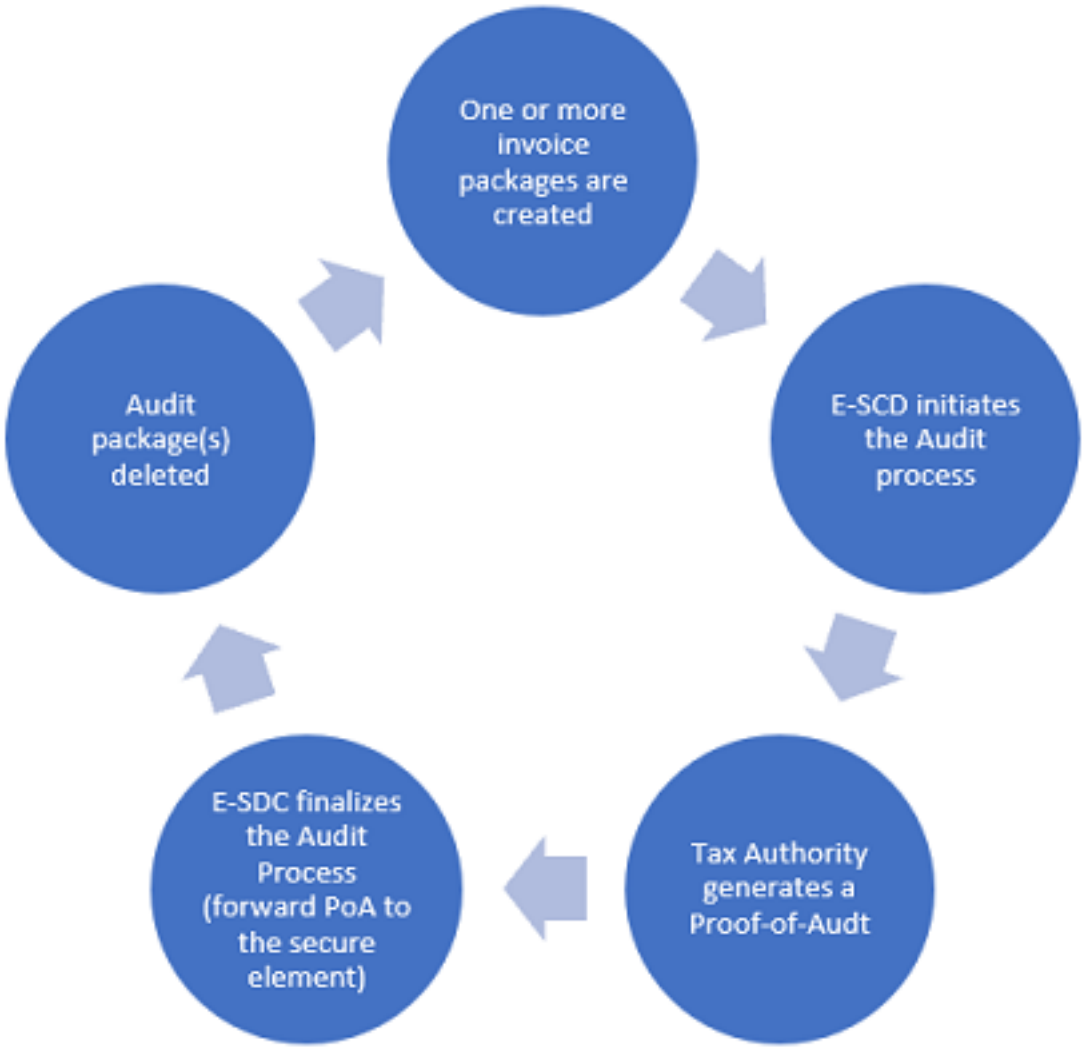
between two Audit Starts should be shortened to 10 minutes (unless there were no new invoices created in that period).

NOTE:
Although the Audit Start should be initiated periodically, audit packages should be submitted continuously, as soon as they are created (regardless of the Audit Start and Audit End).

The СУФ Развој system needs about 10 minutes to process submitted data and generate a Proof-of-Audit command, provided that there are no missing audit packages for that Audit.

Once the Secure Element receives a valid Proof-of-Audit, the E-SDC can delete the audit packages included in that Audit.

NOTE:
E-SDC must store the SDC number of the last invoice created before an Audit Start (the last invoice included in that Audit) so it would know which invoices to delete upon receiving a valid Proof-of-Audit.



Be mindful of these cases

Sometimes, audit packages can arrive in СУФ Развој database after the Start Audit command - in that case, the system will again need at least 10 minutes after the arrival of the last audit package to generate the End Audit command (**see case 2a below**).

If two Audit Starts are initiated before an Audit End command is generated, the Proof-of-Audit for the first Audit Start will not be valid. Only the next Proof-of-Audit (covering both Audit Starts) will be valid (**see Case 3 below**). Moreover, the Proof-of-Audit for the first Audit Start will be replaced by the next Proof-of-Audit, so E-SDC will receive only the latest Proof-of-Audit. But if E-SDC happens to receive the first Proof-of-Audit, it will be invalid and rejected by the Secure Element.

If one or more audit packages issued by the same secure element (same UID) do not arrive in the database, the End Audit command (Proof-of-Audit) will never be generated.

Audit cycle cases

This means that there are three possible scenarios for completing the Audit cycle:

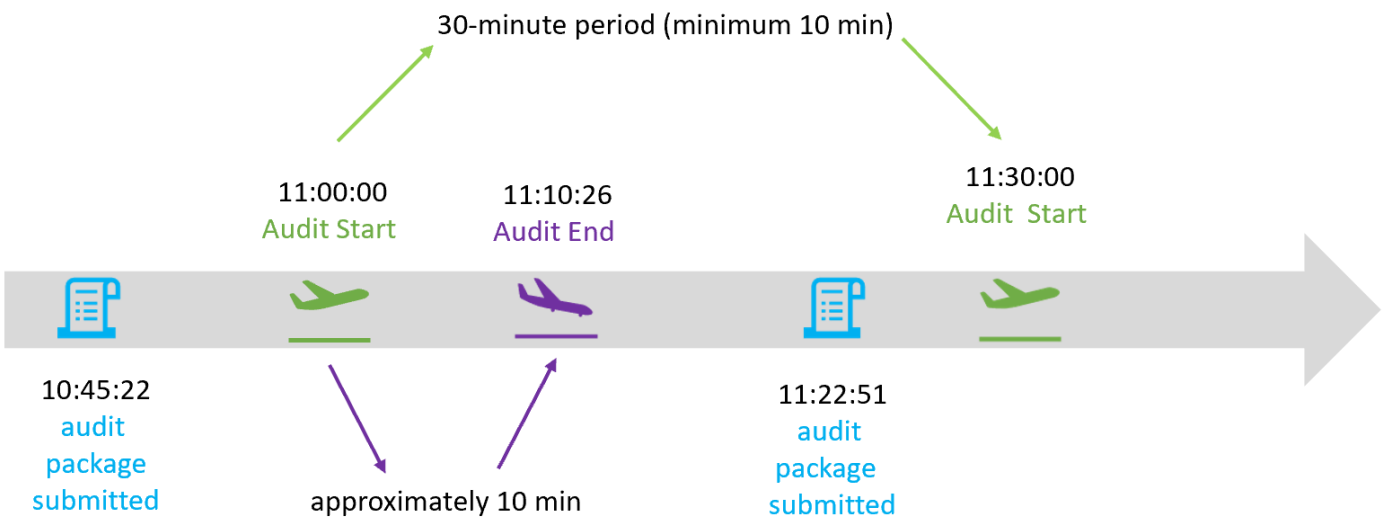
1. One audit package is created and one Audit Start is initiated between completing two Audit Ends (two Proof-of-Audits) - **see Case 1 below**
2. Multiple audit packages are created and one Audit Start is initiated between two Audit Ends (two Proof-of-Audits) - **see Case 2 below**
    - Sometimes, an E-SDC can initiate an Audit Start before submitting all the audit packages from that Audit. In that case, the system will wait for the last audit package from that Audit to arrive before it starts generating the Audit End command (Proof-of-Audit) - **see Case 2a below**
3. Multiple audit packages are created and multiple Audit Starts are initiated between two Audit Ends (two Proof-of-Audits) - **see Case 3 below**

Case 1 – Audit is performed after the creation of each audit package

This is the simplest case, where no additional audit packages are generated during the whole audit process, as follows:

1. Create an audit package
2. Initiate the Audit process by invoking the Start Audit APDU command
3. Receive a proof of audit and pass it to the End Audit APDU command
4. If EndAudit returns the value "true", you can safely delete the audit package(s)
5. If EndAudit returns the value "false", continue until a valid proof of audit is received
6. The period until the next Audit Start must be **at least 10 minutes (recommended from 30 minutes to a couple of hours)** after the previous Audit Start

The figure below illustrates the process:

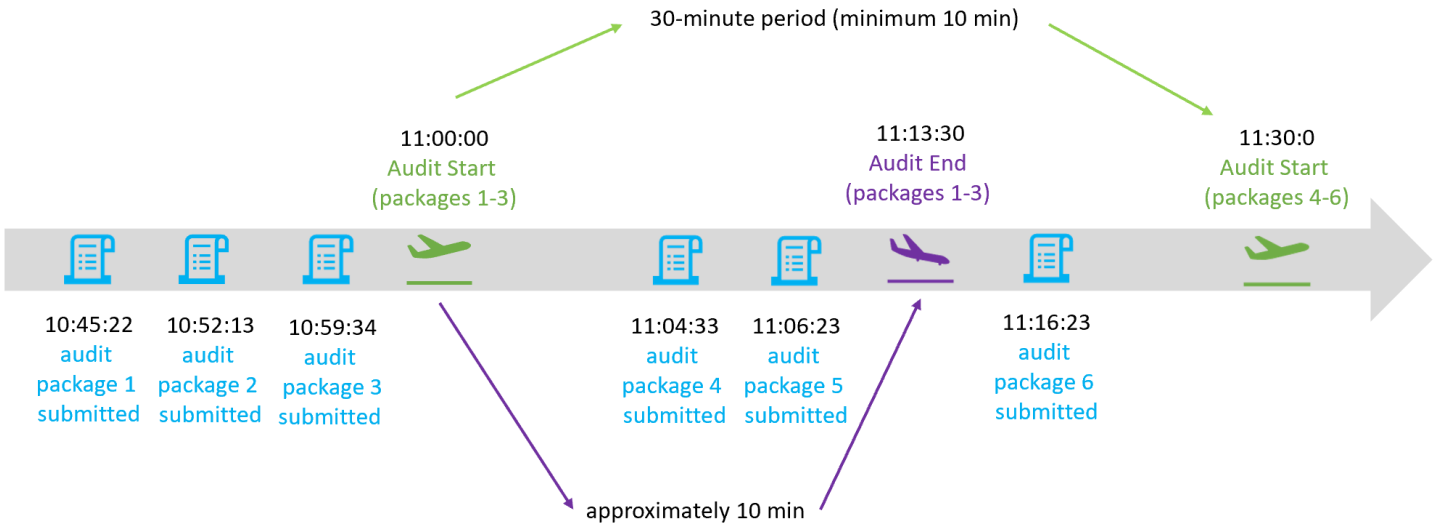


Case 2 – Audit is performed after multiple audit packages have been created

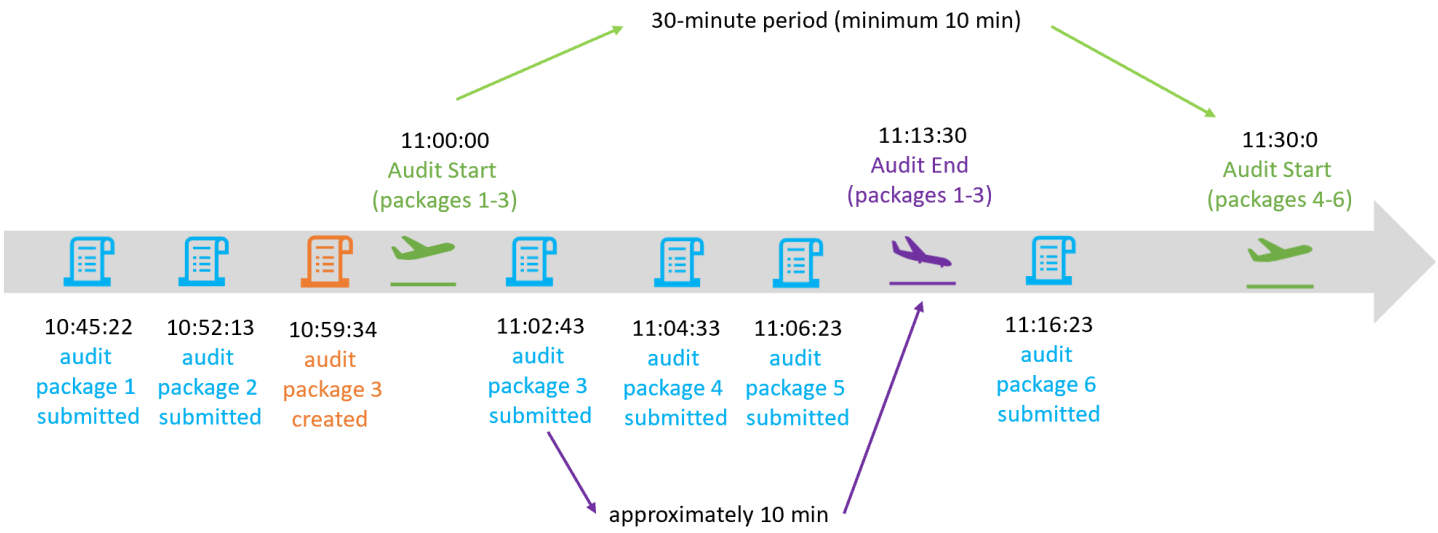
In this case, new packages can be created after an audit start:

1. Create audit packages 1-3 (as shown on the diagram)
2. Initiate the audit process by invoking the Start Audit APDU command
3. Continue to fiscalize invoices and create audit packages 4-6
4. Receive a proof of audit and pass it to the End Audit APDU command
5. If EndAudit APDU command returns value true you can delete remaining audit packages 1-3 because it is the last initial audit being invoked by E-SDC. Audit packages 4-6 are created after the call to BeginAudit APDU command so they are not audited in this cycle
6. If EndAudit APDU command returns value false, continue (return to point 1) until a valid proof of audit is received
7. The period until the next Audit Start must be **at least 10 minutes (recommended from 30 minutes to a couple of hours)** after the previous Audit Start

The figure below illustrates the process:



Case 2a - Audit Start is initiated before all audit packages are submitted



Case 3 – Audit is started multiple times before the Proof-of-Audit is generated

This case involves multiple audit starts:

1. Create Audit Packages 1-3
2. Initiate the Audit process by invoking the Start Audit APDU command
3. Continue to fiscalize invoices and create Audit Packages 4 and 5
4. Initiate another audit process by invoking the Start Audit APDU command – the previous audit is canceled
5. Receive the Proof-of-Audit and pass it to End Audit APDU command
6. If EndAudit returns value true you can delete remaining audit packages 1-5 because it is the last BeginAudit being invoked by E-SDC.
7. If EndAudit APDU command returns value false, continue until a valid Proof-of-Audit is received
8. A Proof-of-Audit generated for the first Audit Start (Audit 1 below) is not considered valid. Only the Proof-of-Audit which is generated for the last Audit Start (Audit 2 below) is considered valid and will be forwarded to the secure element.
9. The period until the next Audit Start must be **at least 10 minutes (recommended from 30 minutes to a couple of hours)** after the previous Audit Start

The figure below illustrates the process: