Spillemyndigheden

Technical requirements – online casino and betting

This guidance contains a description of the technical requirements applicable for operators, who apply for a license to offer online casino and/or betting in Denmark and for operators, who already have a license to offer online casino and / or betting.

Version 2.2

Content

1. INTRODUCTION 4

1.1 LEGAL BASIS 4
1.2 VERSION 5

2. INTRODUCTION TO THE OVERALL SYSTEM COMPLEX 5

2.1 VISUAL PRESENTATION 6

3. SAFE 7

3.1 REQUIREMENTS FOR ACCESSIBILITY AND CONNECTION TO SAFE 7
3.2 REQUIREMENTS FOR STORING AND BACKUP OF DATA 8
3.3 REQUIREMENTS FOR FOLDER STRUCTURE ON SAFE 8
3.3.1 Naming of standard records and zip files 10
3.4 REQUIREMENTS FOR COMPRESSING DATA ON SAFE 10
3.4.1 Placing of zip file in date folder 11
3.5 REQUIREMENTS FOR REPORTING OF GAME DATA 12
3.6 THE DGA'S PROCESS FOR COLLECTING DATA 12
3.7 SAFE DURING THE LICENSE APPLICATION PROCESS 13
3.8 CHANGES TO AND REPLACEMENT OF SAFE 14

4. TAMPERTOKEN 14

4.1 TECHNICAL REQUIREMENTS RELATED TO TAMPERTOKEN 15
4.1.1 Guidance and examples of using services 16
4.1.2 Error handling for TamperToken services 19
4.1.3 Handling of unused tokens 19
4.2 MECHANISM FOR MAC GENERATION 20
4.2.1 MAC API 22
4.2.2 Example of MAC calculation 23

5 ROFUS – REGISTER OF SELF-EXCLUDED PLAYERS 23

5.1 TECHNICAL REQUIREMENTS RELATED TO ROFUS 24
5.1.1 Guidance and examples of using services 24
5.1.2 Enquiry in ROFUS upon account opening and account login 26
5.2 "NO THANK YOU TO GAMBLING COMMERCIALS" IN ROFUS 29
5.2.1 Guideline to mass requests in ROFUS (No thank you to gambling commercials) 29
5.2.2 Service call and CPR numbers 30

6 ACCESS TO AND TEST OF TAMPERTOKEN AND ROFUS 30

6.1 LICENSE APPLICANT'S TEST OF TAMPERTOKEN AND ROFUS 31
6.1.1 End-points for services in the test environment 31
6.1.2 License applicant's connectivity test 32
6.2 TEST EVALUATION 32
6.3 ACCESS TO THE TEST ENVIRONMENT AFTER A LICENSE IS ISSUED 32

7. ADDITION OR CHANGE OF GAMBLING SYSTEM 33

8 LICENSE HOLDER'S OBLIGATIONS TO NOTIFY 33

8.1 NEW GAMES AND CHANGES IN EXISTING OFFER OF GAMES 33

8.1.1 Implementation of new games 33

8.1.2 Changes in the existing offer of games 33

8.1.3 Situations where the DGA's standard records cannot be utilised 34

8.2 OTHER OBLIGATIONS TO NOTIFY 34

APPENDIX 1 35

1. Introduction

The purpose of this document is to describe the technical requirements applicable for operators, who wants to offer, or already have a license to offer online casino and betting. The requirements are described in relation to the systems the Danish Gambling Authority (DGA) uses in supervision of license holders. This covers the license holder's data storage (SAFE), the security system TamperToken and the register of self-excluded persons (ROFUS).

The license holder must develop their gambling system, so it can use the interface of the DGA's systems. This makes it possible for the DGA to handle data and perform supervision to ensure that online gambling is offered in accordance with the legislation. All license holders are required to use the specified interfaces to the DGA's systems specifically developed for supervision purposes and to set up a SAFE, which they give the DGA access to.

The technical requirements are described in detail over the next sections. The requirements are grouped in relation to what system they belong to.

Besides being compliant with requirements described in this document, operators applying for a license to offer online casino and betting in Denmark must also comply with the DGA's certification programme, which can be found on Spillemyndigheden.dk.

1.1 Legal basis

The legal basis for the technical requirements is the executive orders for online casino, land based betting and online betting.

According to section 33 in the executive order on online casino, section 11 in the executive order on land-based betting and section 28 in the executive order on online betting, license holders must comply with the technical requirements, which appear from annex 1 to the executive orders.

The executive orders including annex 1 is available on spillemyndigheden.dk.

Noncompliance with the requirements is punishable.

1.2 Version

Version	Date	Short description of the change
2.1	27.11.2018	This document replaces the following documents: • Obtaining a license v1.2 • Technical requirements v1.11 • Directions for service usage v1.05 • Test af standard records ver 1.0 • ROFUS services - english - 01jul2015 • TamperToken service - 03mar2011 • Gamblermultireklamecheck The version starts with 2.1 so the Danish and English editions have the same version number.
2.2	27.01.2020	Former section 3.4.2 about the size of a zip file has been removed, because it could conflict with complying with the token frequency. Section 5.2 about "no thank you to gambling commercials" has been updated, because the service call is now mandatory. Furthermore, some lingual corrections and precisions have been made.

2. Introduction to the overall system complex

The system complex consists of the license holder's gambling system, the license holder's data storage (SAFE), a security system (TamperToken) and the register of self-excluded players (ROFUS).

SAFE is the license holder's own data storage (a secure FTP file server), where the license holder stores data for all games completed in the license holder's gambling system used for their Danish offer. All license holders must establish a SAFE and give the DGA access. Game data must comply with the requirements described in "Guidance on reporting of games".

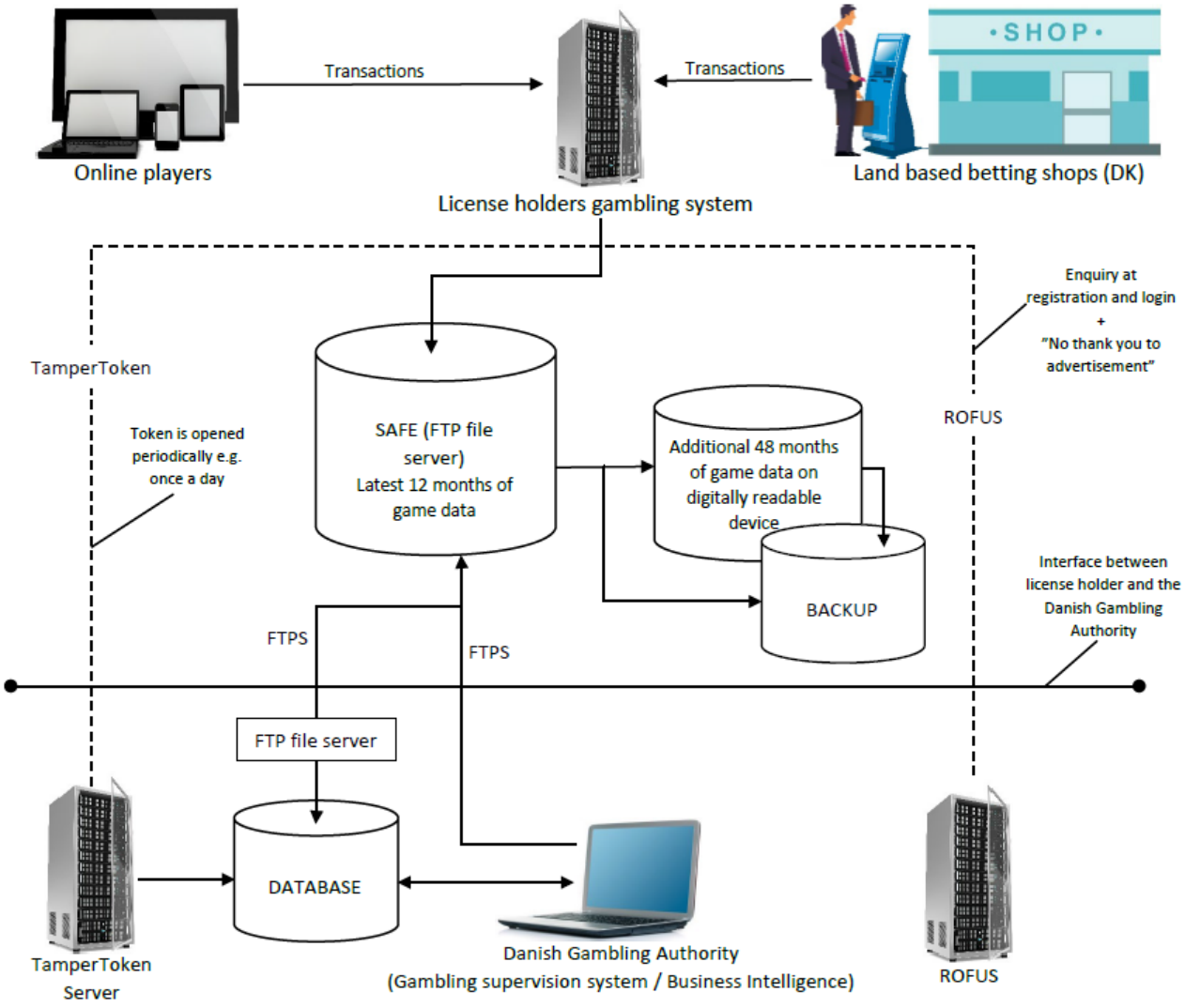
TamperToken is a security system with the purpose to ensure that data saved in the license holder's SAFE remains unchanged, while stored by the license holder.

ROFUS is a register of players in Denmark, who have voluntarily excluded themselves temporarily or permanently from being able to play online gambling in Denmark. The register is located with the DGA, who is responsible for keeping the system.

In combination these three systems forms the technical foundation making it possible for license holders to offer online gambling in Denmark legally.

2.1 Visual presentation

The complete system complex can be illustrated like this:



3. SAFE

In order to get a license for offering online casino and betting, a data storage (SAFE) must be established. The license holder must use the SAFE for reporting game data.

The license holder establishes the SAFE. The license holder can use a supplier for operating their SAFE, but the license holder is at all time legally responsible for the operation of the SAFE and the data stored in the SAFE.

3.1 Requirements for accessibility and connection to SAFE

1. SAFE shall be on a server physically separated from the license holder's gambling system. The servers can be located in the same data center.

2. Data stored in SAFE must be logically and safely separated from any other data.

3. The license holder must ensure that the DGA has online access to retrieving game data from SAFE. There shall be a guaranteed uptime of at least 98.5 % measured pr. month.

4. SAFE shall be configured in UTC time, so time stamps on files[1] and folders are stated in UTC time.

5. Transmission of data shall happen using the internet with FTPS/Implicit SSL in passive mode on port 990. Reuse of SLL connection must not be used. License holders shall establish a sufficient connection to ensure an unproblematic transmission of data.

6. To make it possible for the DGA to access the SAFE using FTPS, the license holder shall place a certificate on the FTPS connection. The certificate must be issued by a Certificate Authority.

7. To make it possible for the DGA to connect to the SAFE, the license holder must whitelist these IP addresses:
   - 84.255.109.80
   - 84.255.109.85
   - 84.255.109.86
   - 91.230.68.13
   - 91.230.68.190
   - 185.151.193.16
   - 185.151.193.17
   - 185.151.193.18
   - 185.151.193.19
   - 185.151.193.20
   - 185.151.193.21
   - 185.151.193.22
   - 185.151.193.23
   - 194.239.239.10
   - 194.239.239.30
   - 194.239.239.31
   - 194.239.239.32
   - 194.239.239.33
   - 194.239.239.34

8. The DGA must be able to access SAFE using FTPS/Implicit SSL in passive mode on port 990. A port range between 40.000 and 50.000 must be used. The license

[1] For specific requirements see "Guidance on reporting of game data" on spillemyndigheden.dk

holder can use a smaller port range as long as it is within these two limits. TLS-resuming must not be activated on the FTP server.

3.2 Requirements for storing and backup of data

The DGA shall have online access to the latest 12 months of game data. Additional 48 months of game data shall be stored on a digitally readable device. The license holder must, on request, be able to deliver archived game data from a digitally readable device to the Danish Gambling Authority within 5 working days.

The license holder shall ensure necessary backup of all data. SAFE and backup of SAFE must be geographically separated. In addition, the data stored on a digitally readable device shall be geographically separated from the backup of the data thus stored.

"Geographically separated" means that the servers used for SAFE and backup SAFE must not be located at the same datacenter.

3.3 Requirements for folder structure on SAFE

The license holder shall configure the SAFE based on this folder structure, which also appear from section E.3 in annex 1 to the executive orders.

The name of the folders is case-sensitive so the exact name must be used:

Level 1: The outermost folder should be "folderstruktur-spilsystem".

Level 2: At this level there is only one folder named "Zip".

Level 3: At this level there is a folder for each day named after the date using the format YYYY-MM-DD.

Level 4: At this level there are a number of zip-files, where each is connected to one token. There are also folders for the tokens which are not yet closed. A folder which has not been closed yet is named SpilCertifikatIdentifikation-TamperTokenID. The zip-file which contains the folder is named SpilCertifikatIdentifikation-TamperTokenID.zip.

Level 5: At this level the folders, which each zip-file contains, are placed. They are named: "EndOfDay", "FastOdds", "Jack- pot", "KasinoSpil", "Managerspil", "PokerCash- Games", "PokerTurnering" and "Puljespil".

Level 6: At this level there are folders for the relevant dates, named after the date in the format YYYY-MM-DD. Each Standard Record is placed on this level or level 7, and is placed in the folder that matches the time where the file is created. The date is found in

"TamperTokenUdstedelseDatoTid" from response of Tamper-TokenHent.

Level 7 (optional): There is a possibility for creating subfolders containing time intervals in the format HH.MMHH-MM.

Visual presentation of the folder structure:

3.3.1 Naming of standard records and zip files

Both standard records and zip files on SAFE must follow this naming:

Standard records are named this way:
SpilCertifikatIdentifikation-TamperTokenID-SequenceInToken.xml (Standard records shall be stored as xml-files)

Zip files are named this way:
SpilCertifikatIdentifikation-TamperTokenID.zip

Explanation for naming:

SpilCertifikatIdentifikation:	Text string provided by the DGA to the license holder during the connection process. This will be equal to the username, which the license holder is granted for the TamperToken system.
TamperTokenID:	Identification number of each tamper token. The ID is received by calling the service operation TamperTokenHent in the service TamperTokenAnvend.
SequenceInToken:	A serial number, starting from 1 and ending with E for "End" (1, 2, 3,…,E) to indicate the sequence of which each standard record is included in the MAC algorithm for each token. It is the license holder's responsibility to build a mechanism for generating the sequence.

Example:
SpilCertifikatIdentifikation = SpilApS
TamperTokenID = 1234567
SequenceInToken = 3

The stancard record will have this name: SpilApS-1234567-3.xml.
The Zip-file containing the standard record will have the name SpilApS-1234567.zip.

3.4 Requirements for compressing data on SAFE

To save disk space for the license holder and to simplify the transfer of files, the standard record files must be compressed in a zip file in a continuous process, when they appear on SAFE. The compression must be performed the following way:

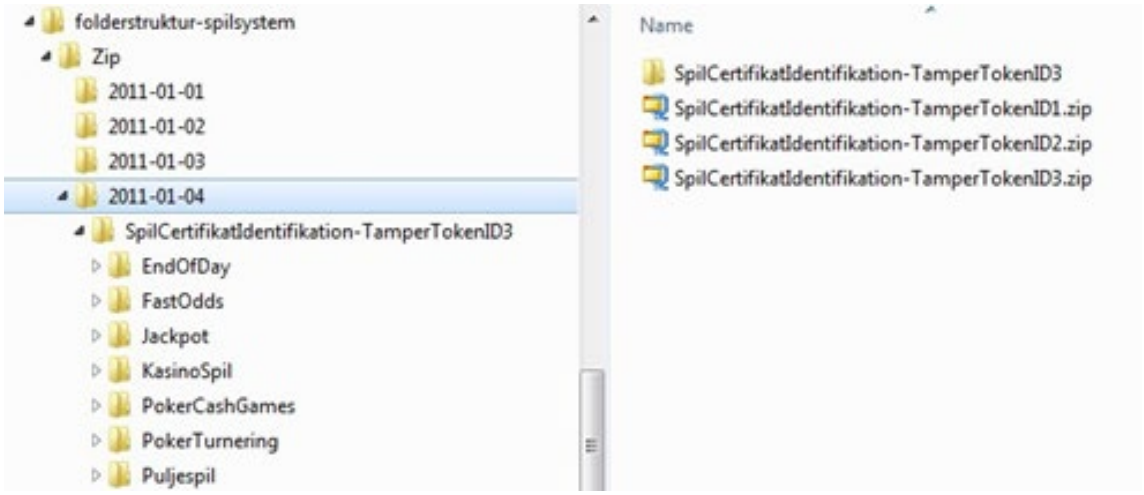
When a standard record file is reported to the SAFE, the following should happen:

1. The MAC algorithm is to be run for each xml file, as described in section 4.2 – Mechanism for generating of MAC value.
2. The standard record is saved in the folder structure for the present token.
3. The standard record is added to the zip-file for the present token.

When the token is closed and all standard record files are added to the zip-file, the folder, which matches the zip-file, should be deleted. The license holder is responsible for developing a mechanism to ensure, that these three steps are performed correctly.

The steps above can be illustrated by this example:
On the figure below it is illustrated that the folder 2011-01-04 has two closed tokens, which is associated to the zip-files: SpilCertifikatIdentifikation-TamperTokenID1.zip and SpilCertifikatIdentifikation-TamperTokenID2.zip, and one open token which is associated to SpilCertifikatIdentifikation-TamperTokenID3.zip.

It shows, that SpilCertifikatIdentifikation-TamperTokenID3.zip is open, since there is both a zip-file and a folder with the same name. The standard records reported continuously and is associated to token 3 should be saved in the folder SpilCertifikatIdentifikation-TamperTokenID3 and appended to SpilCertifikatIdentifikation-TamperTokenID3.zip. When token 3 is closed and all standard records are added to the zip file, the folder SpilCertifikatIdentifikation-TamperTokenID3 should be deleted.



3.4.1 Placing of zip file in date folder

As described in the section above, the zip-file must be placed under level 3 in the folder structure on SAFE. Level 3 contains folders with dates, and the zip-file must be placed under the correct date.

The zip-file must be placed in the folder representing the date corresponding with the date, when the token is opened. This date is found in the response from the service operation TamperTokenHent and is the first 10 characters of the element TamperTokenUdstedelseDatoTid.

In the example below, the date is found in the value **2011-10-16T15:21:19.221+02:00** in the data element **TamperTokenUdstedelseDatoTid**. The zip-file, which is build using this token, must therefore be found on SAFE in the folder: folderstruktur-spilsystem/Zip/2011-10-16/

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Header/>
  <env:Body>
    <ns:TamperTokenAnvend_O xmlns:ns="http://skat.dk/begrebsmodel/2009/01/15/">
      <ns:Kontekst>
        <HovedOplysningerSvar xmlns="http://skat.dk/begrebsmodel/xml/schmas/kontekst/2007/05/31/">
          <TransaktionsID>895ffb40-9f4a-11e0-8264-0800200c9a66</TransaktionsID>
          <ServiceID>TamperTokenAnvendService</ServiceID>
          <TransaktionsTid>2011-06-25T18:41:30.054+01:00</TransaktionsTid>
        </HovedOplysningerSvar>
      </ns:Kontekst>
      <ns:TamperTokenHent_O>
        <ns:TamperTokenID>1234567</ns:TamperTokenID>
        <ns:TamperTokenStartMAC>91c5e2c0e033e3b18fc66bfa43bb08d4</ns:TamperTokenStartMAC>
        <ns:TamperTokenUdstedelseDatoTid>2011-10-16T15:21:19.221+02:00</ns:TamperTokenUdstedelseDatoTid>
        <ns:TamperTokenPlanlagtLukketDatoTid>2011-10-17T15:21:19.221+02:00</ns:TamperTokenPlanlagtLukketDatoTid>
  </ns:TamperTokenHent_O>
 </ns:TamperTokenAnvend_O>
</env:Body>
 </env:Envelope>
```

3.5 Requirements for reporting of game data

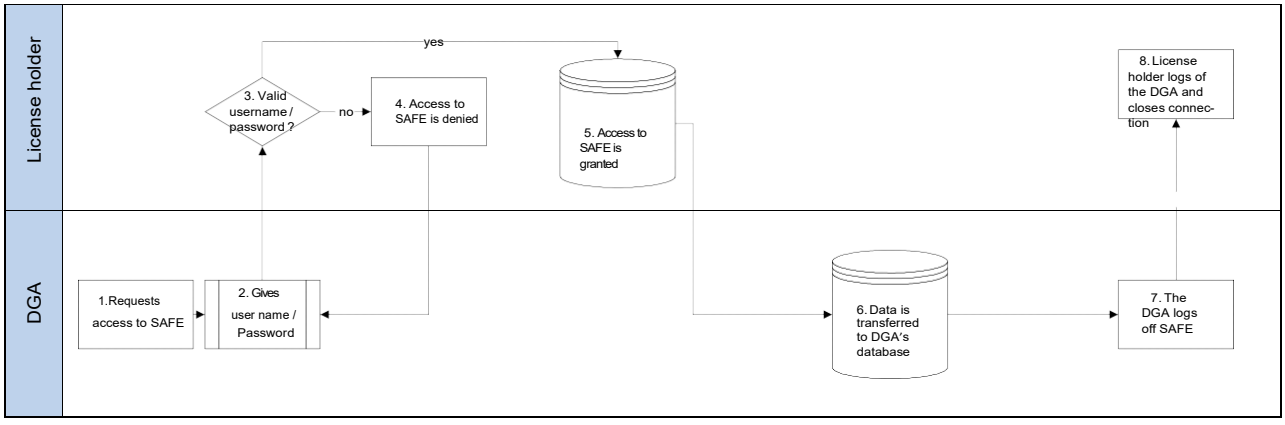
To make it possible for the DGA to load game data reported by the license holder, reporting must be done using the standard records specified by the DGA.

The requirements for standard records are described in "Guidance on reporting of game data" available at spillemyndigheden.dk.

3.6 The DGA's process for collecting data

The purpose of this section is to give the license holder an insight to the DGA's process for collecting data from the license holder's SAFE. License holders must make it possible for the DGA to collect data from their SAFE in accordance with the following process:

Flow chart:



Process card	
Process stakeholders	License holders and the DGA
Purpose of the process	The purpose of the process is to ensure that the DGA can collect data from the license holder's SAFE for supervision.
Process interface	FTPS/Implicit SSL in passive mode on port 990
Input (start)	The process starts when the DGA requests access to SAFE using the user name and password issued by the license holder.
Output (end)	The process ends when the DGA has collected the requested data and has logged off SAFE.

Description of process flow	
Number	Description
1	The DGA requests access to SAFE
2	The DGA fills in its user name and password
3	The license holder's system validates user name and password
4	If the user name and password does not validate, access is denied and the DGA is returned to step 2.
5	If the user name and password are valid, the system grants access to viewing data in SAFE and download may be commenced.
6	Data are transferred to the DGA's server
7	The DGA logs off SAFE
8	SAFE logs off the DGA

3.7 SAFE during the license application process

During the license application process the following steps must be completed:

1. The applicant fills out the section regarding SAFE in appendix B to the application. This consists of: Username, password, IP-address and possibly URL, and is used by the DGA to establish connection to SAFE. During the application process the DGA tests the connection in cooperation with the applicant.

2. The applicant must deliver test data to the DGA. Test data is reported using SAFE and the TamperToken test environment, so data can be loaded in the DGA's database. See section 7 for information on getting access to the test environment. The requirements for test data for each game category appears from annex 1 to this document. The test data can be reported in connection with completion of TamperToken test case cf. section 7.1.

   When test data has been reported and loaded in the DGA database, the DGA performs a review of the data, so it is ensured that data complies with the requirements.

3. The applicant must fill out a document (description of attributes), in which the applicant in their own words describes the content of each data attribute in the standard records used for reporting of game data. The document is send to the applicant during the application process. The DGA performs an evaluation of the descriptions, and any uncertainties are clarified in dialogue with the applicant.

3.8 Changes to and replacement of SAFE

If a license holder wants to make changes to their existing SAFE or wants to replace their existing SAFE the DGA must be informed in reasonable time ahead of the change or replacement. There needs to be time to make the necessary changes to ensure that connection to SAFE is maintained after the change or replacement. New IP-addresses for the SAFE needs to be whitelisted in the DGA's system, this can take up to 4 weeks.

In these cases, the DGA makes an assessment of the need for new tests. It is always necessary to take some action in order to create connection between the license holder's new/changed SAFE and the DGA's system.

The license holder must inform the DGA about any changes to the URL, IP address and the username and password to the SAFE. This is necessary for the DGA to setup the system correctly.

The license holder must ensure to whitelist the IP addresses, from where the DGA connects to the SAFE. See section 3.1 for a list of the IP addresses.

4. TamperToken

The DGA uses the security system Tamper Token. The purpose of the Tamper Token system is to ensure that data, i.e. Standard Records, will remain unchanged while they are stored in SAFE at the license holder's end.

TamperToken handles the following functions:

- Creation of keys (tokens) to be used in the calculation of the MAC (Message Authentication Code)
- Storage of MACs for later control
- Continuous control to check that the period of time for terminating tokens is observed. By default, the token frequency is 24 hours, unless the DGA informs otherwise.
- Verifying that a retrieved series of Standard Records has not been altered relative to the calculated MAC

As an introduction to the TamperToken service, this section contains a step-by-step description of the process from opening a token to the token is closed. Details on each step is found in separate sections in this document. References are made where appropriate.

1. Call TamperTokenHent Service (see section 4.1.1)
    a) If the call is not successful, license holder will create an incident to resolve the error and continue to use previous token
    b) If the call is successful, the following information will be returned in the response: TamperTokenID, TamperTokenStartMAC, TamperTokenPlanlagtLukketDatoTid and TamperTokenUdstedelseDatoTid

2. For the first Standard Record file, the TamperTokenStartMAC is used to generate a MAC for this file e.g. SpilCertifikatIdentifikation-TamperTokenID-SequenceInToken.xml (see section 4.2 for MAC mechanism and section 3.3.1 for naming of standard record)

- SpilCertifikatIdentifikation is "Username for TamperToken system"
- TamperTokenID is result from TamperTokenHent
- SequenceInToken is a sequential number from 1 to E ("E" for End when closing token)

3. When new standard record files are generated, the MAC from the previous file is used to generate a new MAC for the next file (see section 4.2.1)

4. After MAC generation the standard record file (xml) is added to a Zip-file and folder for the present token e.g. SpilCertifikatIdentifikation-TamperTokenID.zip (file) and SpilCertifikatIdentifikation-TamperTokenID (mappe) (see section 3.4 for placement of data on SAFE)

- SpilCertifikatIdentifikation is "Username for TamperToken system"
- TamperTokenID is result from TamperTokenHent

5. Continue to store standard record files on SAFE (see section 3.3)

6. Perform "step 1" above to open a new token before moving on to "step 7", where the current token is closed. This way the license holder always has an open token for reporting data.

7. After the time interval given from TamperTokenHent (TamperTokenPlanlagtLukketDatoTid), the license holder calls TamperTokenLuk service to close the token (see section 4.1.1)

a) The call is provided with the TamperTokenID of the token the license holder wants to close, the SpilCertifikatIdentifikation and the latest generated MAC
    - TamperTokenID is result from TamperTokenHent
    - SpilCertifikatIdentifikation is "Username for TamperToken system"
b) If the call is not successful, the license holder will create an incident to resolve the error and begin to use the new token (opened en step 6) for reporting data

8. When the token is closed, the license holder deletes the SpilCertifikatIdentifikation-TamperTokenID folder. Content of this folder will be in the SpilCertifikatIdentifikation-TamperTokenID.zip file.

4.1 Technical requirements related to TamperToken

License holder must implement the TamperToken solution, which must be used when reporting gambling data.

See section 6 regarding access to the TamperToken test environment.

4.1.1 Guidance and examples of using services

The DGA has developed a web service called "TamperTokenAnvend", which contains of 2 operations:

1. TamperTokenHent
   The operation must be used when the license holder has to retrieve a token. The operation TamperTokenHent returns a generated key (TamperTokenStartMac), which must be used by the license holder to generate a MAC (Message Authentication Code). See section 4.2.

2. TamperTokenLuk
   The operation must be used when the license holder has to close a token after the data has been compressed in a zip file on SAFE. The operation returns a receipt with an approval or a message error.

4.1.1.1 Header information in service calls

When making a service call, header information must be stated. The purpose of the header information is to be able to follow request and response for service calls, and to be able to report wrong information.

Header- and error information is handled identically for TamperToken and ROFUS services. The information below can therefore also be found in the section regarding ROFUS.

The header information is inserted in an any-element on each service and must comply with the format specified in the XSD-files for header information, which are found on spillemyndigheden.dk.

Header information in "request":
The following header information must be stated in a service request made by the license holder:

- TransaktionsID: License holder must generate a unique transaction id for the service call. The DGA recommends to follow the standard Universally Unique Identifier (UUID), where the id consists of 32 hexa decimals represented in 5 groups separated by dashes on the form 8-4-4-4-12. E.g.: 07B2A963-26C4-47E0-B517-C7059A598DA3

- TransaktionsTid: The time of transaction. The time must be stated on the form YYYY-MM- DDThh:mm:ss.sTZD, where YYYY is year, MM is month, DD is day, hh is hours, mm is minutes, ss is seconds, s is one or more digits for seconds, and TZD is the time zone represented as Z or +hh:mm or –hh:mm. E.g.: 2010-12-07T09:33:51.249+01:00.

Header information in "response":
The following header information is always stated in a service response:

- TransaktionsID: Same as above.
- TransaktionsTid: Same as above.
- ServiceID: The name of the called service.

The following header information is also stated in a service response, but is only stated when necessary.

- Fejl: Errors are reported when a request is not completed as expected.
  - FejlNummer: Id-number for the error.
  - FejlTekst: Description in text of the error.
  - Identifikation: Text code for the error.
  - ServiceID: Same as above.

- Advis: Notifications are messages which are not errors. It could be a message explaining that the service call has been executed as expected.
  - AdvisNummer: Id-number for the notification.
  - AdvisTekst: Description in text of the notification.
  - Identifikation: Text code for the notification.
  - ServiceID: Same as above.

4.1.1.2 Examples of service calls

The DGA has developed two examples of service calls. The examples shows how you, in respectively Java and .Net, can get web service descriptions and call services by the use of HTTP basic access authentication. Furthermore, it is shown how data can be received from the service. The service GamblerCheck is used in the example.

The DGA have developed following example files, also to be found on spillemyndigheden.dk:

- Example in .Net: GamblerServiceExampleClient.cs
- Example in java: GamblerServiceExampleClient.java

Besides those examples, the following contains a number of examples on service request and response for the service calls TamperTokenHent and TamperTokenLuk. The license holder must make these calls in order to open and close a token. The examples are meant as a help to the license holder's understanding of the service calls, but it is not the intention that the license holders can write code based on the examples. For this purpose, we refer to XSD schemas and WSDL files.

Example of TamperTokenHent:

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns="http://skat.dk/begrebsmodel/2009/01/15/">
  <soapenv:Header/>
  <soapenv:Body>
    <ns:TamperTokenAnvend_I>
      <ns:Kontekst>
        <ns1:HovedOplysninger xmlns:ns1="http://skat.dk/begrebsmodel/xml/schemas/kontekst/2007/05/31/">
          <ns1:TransaktionsID>895ffb40-9f4a-11e0-8264-0800200c9a66</ns1:TransaktionsID>
          <ns1:TransaktionsTid>2011-06-25T18:41:30.054+01:00</ns1:TransaktionsTid>
        </ns1:HovedOplysninger>
      </ns:Kontekst>
     <ns:TamperOperationValg>
       <ns:TamperTokenHent>
          <ns:SpilCertifikatIdentifikation>TamperTokenTest3</ns:SpilCertifikatIdentifikation>
        </ns:TamperTokenHent>
      </ns:TamperOperationValg>
    </ns:TamperTokenAnvend_I>
```

```
    </soapenv:Body>
  </soapenv:Envelope>
```

Response:

```
  <env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
    <env:Header/>
    <env:Body>
      <ns:TamperTokenAnvend_O xmlns:ns="http://skat.dk/begrebsmodel/2009/01/15/">
        <ns:Kontekst>
          <HovedOplysningerSvar xmlns="http://skat.dk/begrebsmodel/xml/schemas/kontekst/2007/05/31/">
            <TransaktionsID>895ffb40-9f4a-11e0-8264-0800200c9a66</TransaktionsID>
            <ServiceID>TamperTokenAnvendService</ServiceID>
            <TransaktionsTid>2011-06-25T18:41:30.054+01:00</TransaktionsTid>
          </HovedOplysningerSvar>
        </ns:Kontekst>
        <ns:TamperTokenHent_O>
          <ns:TamperTokenID>1234567</ns:TamperTokenID>
          <ns:TamperTokenStartMAC>a06174fd062bb397894860bd5c20aa08</ns:TamperTokenStartMAC>
          <ns:TamperTokenUdstedelseDatoTid>2011-06-25T18:47:04.481+02:00</ns:TamperTokenUdstedelseDatoTid>
          <ns:TamperTokenPlanlagtLukketDatoTid>2011-06-26T18:47:04.481+02:00</ns:TamperTokenPlanlagtLukketDatoTid>
        </ns:TamperTokenHent_O>
      </ns:TamperTokenAnvend_O>
    </env:Body>
  </env:Envelope>
```

Example of TamperTokenLuk:

Request:

```
  <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ns="http://skat.dk/begrebsmodel/2009/01/15/">
   <soapenv:Header/>
     <soapenv:Body>
       <ns:TamperTokenAnvend_I>
         <ns:Kontekst>
           <ns1:HovedOplysninger xmlns:ns1="http://skat.dk/begrebsmodel/xml/schemas/kontekst/2007/05/31/">
             <ns1:TransaktionsID>895ffb40-9f4a-11e0-8264-0800200c9a66</ns1:TransaktionsID>
             <ns1:TransaktionsTid>2011-06-25T18:41:30.054+01:00</ns1:TransaktionsTid>
           </ns1:HovedOplysninger>
         </ns:Kontekst>
         <ns:TamperOperationValg>
           <ns:TamperTokenLuk>
             <ns:TamperTokenID>1234567</ns:TamperTokenID>
             <ns:SpilCertifikatIdentifikation>TamperTokenTest3</ns:SpilCertifikatIdentifikation>
             <ns:TamperTokenMAC>2da9fe732840bc40f05eefbace7bf03fc36e141907a8d6ce7da329fa0f1bb25c
             </ns:TamperTokenMAC>
           </ns:TamperTokenLuk>
         </ns:TamperOperationValg>
       </ns:TamperTokenAnvend_I>
     </soapenv:Body>
  </soapenv:Envelope>
```

Response:

```
  <env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
    <env:Header/>
    <env:Body>
      <ns:TamperTokenAnvend_O xmlns:ns="http://skat.dk/begrebsmodel/2009/01/15/">
        <ns:Kontekst>
          <HovedOplysningerSvar xmlns="http://skat.dk/begrebsmodel/xml/schemas/kontekst/2007/05/31/">
```

```
            <TransaktionsID>895ffb40-9f4a-11e0-8264-0800200c9a66</TransaktionsID>
            <ServiceID>TamperTokenAnvendService</ServiceID>
            <TransaktionsTid>2011-06-25T18:41:30.054+01:00</TransaktionsTid>
            <SvarReaktion>
              <Advis>
                <AdvisNummer>0</AdvisNummer>
                <AdvisTekst>Token is now closed</AdvisTekst>
                <ServiceID>TamperTokenAnvendService</ServiceID>
              </Advis>
            </SvarReaktion>
          </HovedOplysningerSvar>
        </ns:Kontekst>
      </ns:TamperTokenAnvend_O>
    </env:Body>
</env:Envelope>
```

4.1.2 Error handling for TamperToken services

TamperTokenHent:

If a license holder cannot collect a new token, before the token, which already is in use, expires, the license holder should continue to report data with the open token, even though this token cannot be closed on time.

If the license holder cannot correct the error by themselves, the DGA must be notified.

Once the error is corrected, the license holder can get a new token, and close the old one immediately after.

TamperTokenLuk:

If a license holder cannot close a token on the planned time, the license holder must begin to report data in the new token, which should be collected immediately before, and then begin to investigate the error.

If the license holder cannot correct the error by themselves, the DGA must be notified.

Once the error is corrected, the license holder can close the token.

It is important, that all data is in place when the token is closed, because the DGA will start copying the data from the license holder's SAFE in the exact moment a token is closed.

4.1.3 Handling of unused tokens

In case a license holder have opened a token with the service TamperTokenHent, which is not be used anyway, the license holder must close this token by using the service operation TamperTokenLuk.

In this situation, the license holder must report the text "empty" in the field TamperTokenMAC, instead of the usually reported calculated MAC value. The service call will look like this:

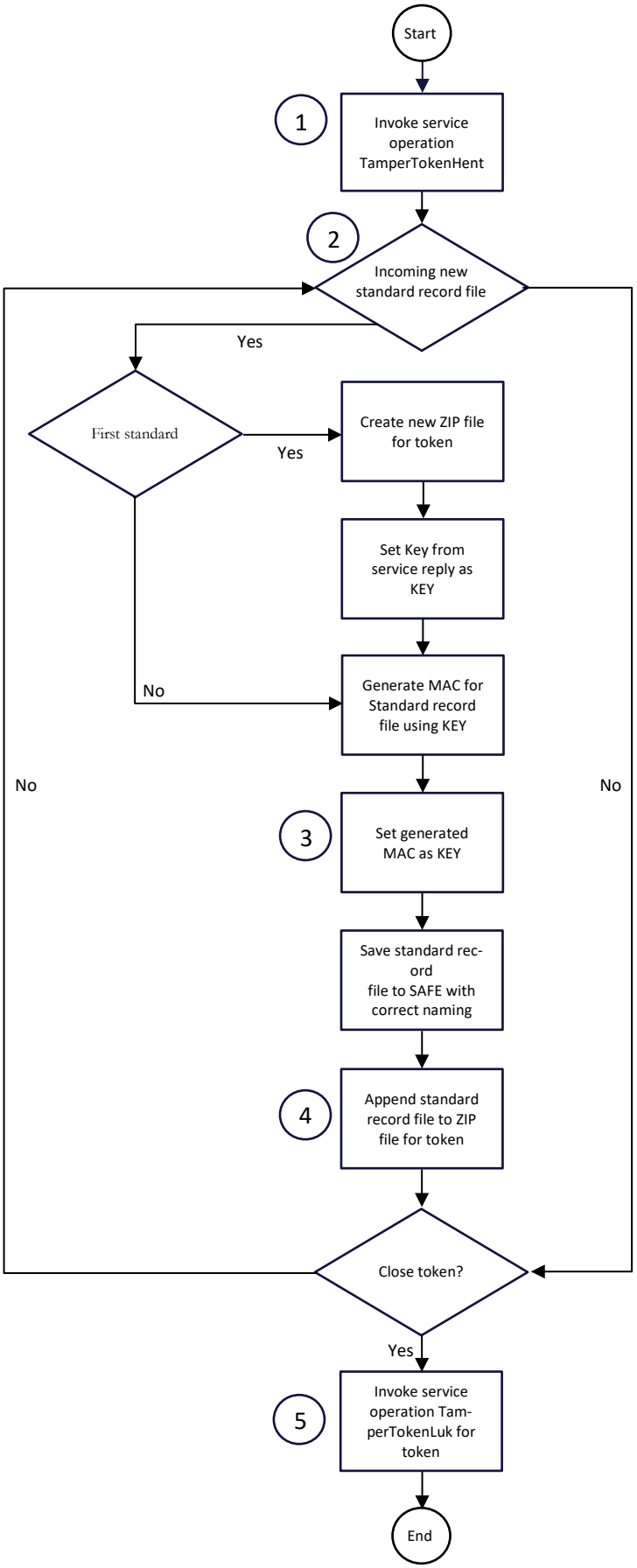
```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns="http://skat.dk/begrebsmodel/2009/01/15/">
    <soapenv:Header/>
    <soapenv:Body>
        <ns:TamperTokenAnvend_I>
                <ns:Kontekst>
                    <ns1:HovedOplysninger xmlns:ns1="http://skat.dk/begrebsmodel/xml/schemas/kontekst/2007/05/31/">
                    <ns1:TransaktionsID>895ffb40-9f4a-11e0-8264-0800200c9a66</ns1:TransaktionsID>
                    <ns1:TransaktionsTid>2011-10-15T18:41:30.054+01:00</ns1:TransaktionsTid>
                </ns1:HovedOplysninger>
            </ns:Kontekst>
            <ns:TamperOperationValg>
                <ns:TamperTokenLuk>
                    <ns:TamperTokenID>1234567</ns:TamperTokenID>
                    <ns:SpilCertifikatIdentifikation>TamperTokenTest3</ns:SpilCertifikatIdentifikation>
                    <ns:TamperTokenMAC>empty</ns:TamperTokenMAC>
                </ns:TamperTokenLuk>
            </ns:TamperOperationValg>
        </ns:TamperTokenAnvend_I>
    </soapenv:Body>
</soapenv:Envelope>
```

4.2 Mechanism for MAC generation

This section contains information about the MAC algorithm and Application Programming Interface (API), that the license holder must build, and that must be used in the process of packing data on SAFE.

The license holder must build a mechanism to generate a MAC in the right way. This MAC must be used in the process of packing data on SAFE.

Illustration for the mechanism of generating a MAC:



Description of process flow	
Number	Description
1	License holder activates the service operation TamperTokenHent and receives a new TamperTokenID and a key (KEY), which must be used for MAC generation for the first standard record file for the new Token.
2	When a new standard record file is generated, a MAC of this record is generated.
3	The generated MAC is now the new KEY for the generation of the next MAC.
4	After the MAC is generated, the present standard record file is added to a ZIP file for the present token.
5	When a token is to be closed, the service operation TamperTokenLuk is activated with ID for the token, the latest generated MAC and the identification of the license holder.

4.2.1 MAC API

For generation of MACs, the class SecretKeySpec from Java 1.8.3 must be applied. Below is an example of how the code can look for the step "Generate MAC for standard record file using KEY".

For the first file, the argument "KEY" is the key from the service operation "TamperTokenHent". For the following file(s) the argument KEY is the MAC from the previous file.

The argument "InputStream" contains the data from a standard record, for which to generate a MAC.

Example:

```
     public String getMAC(String key, InputStream input) throws Tam-
perTokenException {
              try {
                Mac mac = Mac.getInstance("HmacSHA256");
                byte[] byteKey = ByteArrayHandler.parseString(key);
                SecretKeySpec keySpec = new SecretKeySpec(byteKey,
                "HmacSHA256"); mac.init(keySpec);
                byte[] data =
                new
                byte[1024];
                int read;
                while((read=input.read(data)) > -1){
                     mac.update(data, 0, read);
                }
                return ByteArrayHandler.toString(mac.doFinal());
              }
          catch (Exception e) {
                     throw new TamperTokenException(e);
              }
       }
```

4.2.2 Example of MAC calculation

On spillemyndigheden.dk you will find a file called TamperTokenTest3-2152.zip, which has been used below to give an example of calculation of MAC.

The example has been made for SpilCertifikatIdentifikation = TamperTokenTest3 and TamperTokenID = 2152.

The file TamperTokenTest3-2152.zip contains three files:
TamperTokenTest3-2152-1.xml,
TamperTokenTest3-2152-2.xml and
TamperTokenTest3-2152-E.xml.

A start MAC is retrieved by the service operation TamperTokenHent, and it is shown below as TamperTokenStartMAC. The intermediate MACs, which is calculated on each file, is shown afterwards. The MAC from the last file is reported by the service operation TamperTokenLuk in the element TamperTokenMAC.

1. TamperTokenStartMAC =
   fb99919c20c57b01a1ab37fdc576f75a
2. MAC of file TamperTokenTest3-2152-1.xml =
   148f1bc4bfe2be67cfed691f6a703ed90e780f45faab665b5c86a3c8346ad056
3. MAC of file TamperTokenTest3-2152-2.xml =
   a79953be54a71069a07d2d7c63566daaab221de984d93c36ae8c7b26d149df90
4. MAC of file TamperTokenTest3-2152-E.xml =
   1b14a1da76568ab3b96bc64bb7ee02e846fbd7711e3ce40f477b0c66a0663016
5. TamperTokenMAC =
   1b14a1da76568ab3b96bc64bb7ee02e846fbd7711e3ce40f477b0c66a0663016

5 ROFUS – Register of self-excluded players

According to the Executive Order for online casino § 18 and Executive Order for online betting § 17 it must be possible for a player to self-exclude from online gambling in Denmark. This exclusion can be in a temporary basis, where the player excludes himself for a certain period, or it can be a permanent exclusion.

The DGA is responsible for keeping the register of excluded players. Players are able to register from spillemyndigheden.dk.

The register includes data about players, who have excluded themselves from online gambling in Denmark.

Following information is found the register:
- The player's CPR number.
- The date and time of exclusion.
- The date when temporary exclusion ends (only if the exclusion is temporary).

A player who is permanently excluded can at the earliest one year from the date of entry in the register, ask the DGA to be deleted.

In order to fulfill the requirements to the register, the license holders must make some functions available to the players.

The license holder must:

- Inform about the possibility of registration in ROFUS and make access to the register from the website of the license holder
- Check the status of a player in ROFUS when opening an account and at all logins to the account.

See section 6 for information about access to the ROFUS testing environment.

5.1 Technical requirements related to ROFUS

License holder must implement service calls to ROFUS, to make it possible to check a player's exclusion status.

See section 6 regarding access to the ROFUS test environment

5.1.1 Guidance and examples of using services

The following web services must be used in relation to ROFUS:

- GamblerCSRPValidation
  A service to be used to check a player's age prior to account opening. The services also returns an answer whether the player's CPR number exists. This is particularly important since ROFUS does not check whether or not the CPR number exists. This service must always be performed before GamblerCheck (see below).

  See the documents GamblerCSRPValidationRequest.xsd and Gambler CSRPValidationResponse.xsd on spillemyndigheden.dk for content of the service call.

- GamblerCheck
  A service to be used when a player wants to open an account and for each login. This service makes it possible for the license holder to check whether a person is registered in ROFUS, either temporarily, permanently or not at all. This check is made by using the player's CPR number.

  See the documents GamblerCheckRequest.xsd and GamblerCheckResponse.xsd on spillemyndigheden.dk for the content of this service.

5.1.1.1 Header information for service call

When making a service call, header information must be stated. The purpose of the header information is to be able to follow request and response for service calls, and to be able to report wrong information.

Header- and error information is handled identically for TamperToken and ROFUS services. The information below can therefore also be found in the section regarding TamperToken.

The header information is inserted in an any-element on each service and must comply with the format specified in the XSD-files for header information, which are found on spillemyndigheden.dk.

Header information for "request":
The following header information must be stated in a service request made by the license holder:

- TransaktionsID:
  License holder must generate a unique transaction id for the service call. The DGA recommends to follow the standard Universally Unique Identifier (UUID), where the id consists of 32 hexa decimals represented in 5 groups separated by dashes on the form 8-4-4-4-12. E.g.: 07B2A963-26C4-47E0-B517-C7059A598DA3

- TransaktionsTid:
  The time of transaction. The time must be stated on the form YYYY-MM-DDThh:mm:ss.sTZD, where YYYY is year, MM is month, DD is day, hh is hours, mm is minutes, ss is seconds, s is one or more digits for seconds, and TZD is the time zone represented as Z or +hh:mm or –hh:mm. E.g.: 2010-12-07T09:33:51.249+01:00.

Header information for "response":
The following header information is always stated in a service response:

- TransaktionsID: Same as above.
- TransaktionsTid: Same as above.
- ServiceID: The name of the called service.

The following header information is also stated in a service response but is only stated when necessary.

- Fejl: Errors are reported when a request is not completed as expected.
  - FejlNummer: Id-number for the error.
  - FejlTekst: Description in text of the error.
  - Identifikation: Text code for the error.
  - ServiceID: Same as above.

- Advis: Notifications are messages which are not errors. It could be a message explaining that the service call has been executed as expected.
  - AdvisNummer: Id-number for the notification.
  - AdvisTekst: Description in text of the notification.
  - Identifikation: Text code for the notification.
  - ServiceID: Same as above.

5.1.1.2 Examples for service call

The DGA has created two examples of service calls. The examples shows how you, in respectively Java and .Net, can get web service descriptions and call services by the use of HTTP basic access authentication. Furthermore, it is shown how data can be received from the service. The service GamblerCheck is used in the example.

The following example files can be found on spillemyndigheden.dk:

- Example in .Net: GamblerServiceExampleClient.cs
- Example in java: GamblerServiceExampleClient.java

The license holders obtain access to these services via GamblerCheck proxy service. See the documents GamblerCommonTypes.xsd and GamblerService.wsdl on spillemyndigheden.dk to see the content of this service.

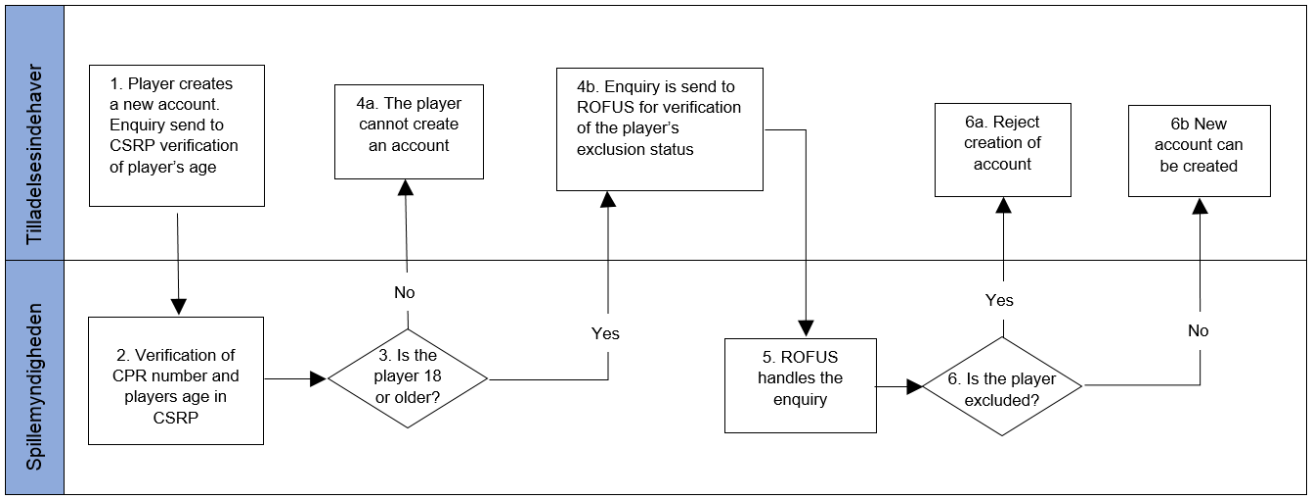
5.1.2 Enquiry in ROFUS upon account opening and account login

In order to gamble on a license holder's website, a player must have a gambling account. New players must open a new gambling account and, and existing players must log on to their gambling accounts before being able to play. The status of the player must be checked in ROFUS in both situations.

5.1.2.1 Enquiry in ROFUS when opening an account

This section describes the process for an enquiry to ROFUS when opening a new gambling account. The process is illustrated by a flow chart and subsequently described step by step in a process card. The purpose is to give precise information about the functionalities the license holder must develop to allow this process to be carried out.

Flow chart:



Process card	
Process stakeholders	License holders and the DGA
Purpose of the process	The purpose of the process is to ensure that the license holder can make enquiries to ROFUS when a player opens a gambling account. The process must be used each time a player wants to open a gambling account at the license holder.
Input (start)	The process starts when the player decides to open a gambling account on the license holder's website.
Output (end)	The process ends when the license holder gets information of the status of the player in ROFUS. If the player is registered temporarily or permanently, the gambling account cannot be opened.

	If the player is not registered in ROFUS, the license holder can continue the opening of the account.

Description of process flow	
Number	Description
1	The player enters the necessary information
2	The player's age is verified via the CSRP. At this stage the existence of the CPR number is also verified. If the CPR number does not exist, the process cannot continue.
3	The CSRP processes the enquiry.
4a	If the player is younger than 18 years, this is reported back to the license holder and the player. Account opening is denied.
4b	If the player is 18 years or older, the license holder will make an enquiry to ROFUS in order to check if the player is registered as excluded.
5	ROFUS processes the enquiry. If ROFUS does not respond, the player can be treated as if he/she is not registered in ROFUS, and the process continues to step 6b. The status of the player must be checked again, once ROFUS is available. If it turns out, that the player is registered in ROFUS, the gambling account must immediately be closed.
6a	If the player is excluded in ROFUS, the opening of an account is denied.
6b	If the player is not excluded, the process of opening an account can continue.

5.1.2.2 Enquiry in ROFUS upon account login

This section describes the process for an enquiry in ROFUS upon login to an existing account. The process is illustrated by a flow chart and subsequently described step by step in a process card. The purpose is to give precise information about the functionalities the license holder must develop to allow this process to be carried out.

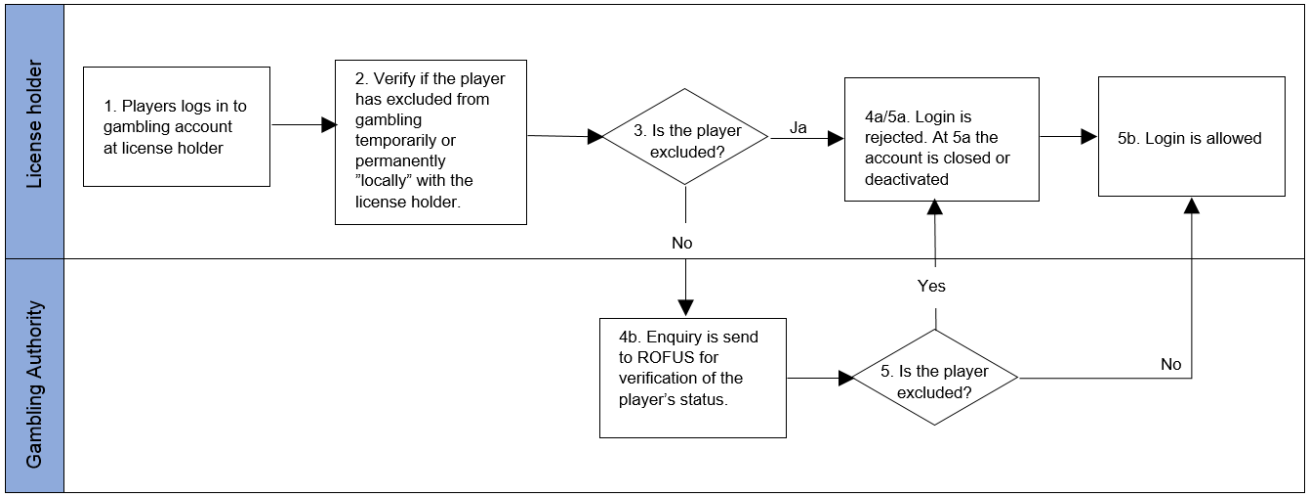
When a player wants to log into an already existing gambling account, the license holder must check whether the player has been registered in ROFUS since his/her last login. The check must be carried out before login is completed. If the player is registered, login is denied.

If the exclusion in ROFUS is temporary, the player's access to the gambling account is denied.

If the exclusion in ROFUS is permanent, the players access to the gambling account is denied, and the license holder must subsequently close the gambling account and terminate the customer relation. Hereafter, the account cannot be reopened. If the player eventually wants to be a customer again, the player must go through the procedure of opening an account once again.

The process of an enquiry in ROFUS upon account login is described below – notice, that the procedure also includes control of the players potential self exclusion directly at the license holder.

Flow chart:



Process card	
Process stakeholders	License holders and the DGA
Purpose of the process	The purpose of the process is to ensure that the license holder can make enquiries in ROFUS when a player logs into his/her gambling account. If the player is registered in ROFUS, the gambling account must be deactivated or closed and the player cannot log in. The process must be used each time a player wants to log into his/her account at the license holder.
Input (start)	The process starts when the player logs into his/her existing account with the license holder
Output (end)	If the player is not registered in ROFUS, the process will end with the player being logged into his/her account. If the player is registered, login will be refused. If the player is temporarily excluded in ROFUS, the player's access to the gambling account will be refused, and the account deactivated. If the exclusion in ROFUS is permanent, the player's access to the gambling account will be refused. The license holder must subsequently close the gambling account and terminate the customer relations.

Description of process flow	
Number	Description
1	The player logs on to his/her account at the license holder
2	The license holder checks in their own system, whether the player is excluded, either on a temporary or on a permanent basis
3	The system of the license holder processes the enquiry.
4a	If the player is excluded locally in the system of the license holder, login to the gambling account is denied. If the exclusion is permanent, the gambling account must be closed and the customer relations must be terminated.
4b	If the player is not excluded locally at the license holder, it is checked with ROFUS, whether the player is excluded, either on a temporary or on a permanent basis.
5	ROFUS processes the enquiry. If ROFUS does not respond, the player can be treated as if he is not registered in ROFUS, and the process continues to step b. The status of the player must be checked again, once ROFUS is available.

5a	If the player is excluded in ROFUS temporarily, the player is denied logon to his/her account, and it the account is deactivated. If the player is excluded permanently, the gambling account must be closed and the customer relations must be terminated.
5b	If the player is not excluded, the player is logged into his/her account.

5.2 "No thank you to gambling commercials" in ROFUS

License holders must implement a service call for ROFUS, to make it possible to check, whether gambling commercials can be sent to a customer.

"Gambling commercial" means any kind of sales contact via telephone numbers, e-mail addresses, post addresses or other information the license holder has about the costumer. Unaddressed household-distributed advertising and internet commercials are not included.

"No thank you to gambling commercials" in ROFUS applies for all gambling commercials from all license holders. In this case, it does not matter what setting the player has in the gambling account in reference to receiving gambling commercials.

All persons, who registers in ROFUS from January 1st 2020, will be covered by "No thank you to gambling commercials". Persons, who have registered before January 1st 2020, had the choice to opt in or not. This means, there can be persons registered in ROFUS, who is not covered by "no thank you to gambling commercials".

5.2.1 Guideline to mass requests in ROFUS (No thank you to gambling commercials)

License holders must make a service call in ROFUS 24 hours at the earliest before sending out gambling commercials to their customers or distributors.

License holders must only call ROFUS with CPR numbers that belongs to people, to whom they have planned to send gambling commercials.

Each service call can be made with 1.000 CPR numbers at the most. This means, that if license holders want to send gambling commercials to 20.000 players, they must make the service call 20 times.

The service call returns CPR-numbers for persons, who may NOT receive gambling commercials.

If ROFUS does not respond, the license holder must check if the error is caused by the license holder's own systems. If that is not the case, the DGA must be notified with information regarding the date and time for the error and the error message.

5.2.2 Service call and CPR numbers

License holders must use this service call:

Input:

GamblerMultiReklameCheck_I

```
(
    *InformationAktørValg*
    [
        TilladelsesindehaverNavn

        * SpillemyndighedBrugerIdentifikation *
        RessourceNummer
    ]
)

*SpillerListe*

0{
    PersonCPRNummer
}
```

Output:

GamblerMultiReklameCheck_O

```
 *SpillerListeReklameFravalgt*

0{
    PersonCPRNummer
}
```

Data element	Data type	Description
PersonCPRNummer	base: string maxLength: 10 pattern: ((((0[1-9]|1[0-9]|2[0-9]|3[0-1])(01|03|05|07|08|10|12))|((0[1-9]|1[0-9]|2[0-9]|30)(04|06|09|11))|((0[1-9]|1[0-9]|2[0-9])(02)))[0-9]{6})|0000000000	The CPR number is a 10 digit number, which uniquely identifies a Danish person.

The service call can be tested in the DGA's test environment for ROFUS. See section 6.3 for access to ROFUS test environment.

6 Access to and test of TamperToken and ROFUS

During the application handling process the license applicant get access to TamperToken and ROFUS test environment. When the DGA has received the application, we create the account and sends the credentials to the applicant.

The DGA does not grant access to the test environment before reception of a formal license application.

Along with the credentials, the DGA sends the test cases, which must be completed and approved as part of the application process.

6.1 License applicant's test of TamperToken and ROFUS

All license applicants must complete the tests mentioned in the test cases, which are send to the applicant during the application process. The applicant must report the received responses in the test case document and add any comments the applicant has.

To get the test result approved the applicant must attach documentation for performing the tests, when returning the test case document. The documentation could be screen dumps or extract from a log file from the applicants gambling system (could be a test/development system).

Any errors experienced when performing the tests must also be reported in the test case document. This information must be added in the comment box for the test, which the error relates to. The test causing the error should be repeated three times to see if it is a general or sporadic error. This information should also be added in the comment box.

After completion of the tests, the results are returned to the DGA in the test case document. The document must be signed. If errors have occurred during the test, these must be resolved by either the license applicant or the DGA, and a new test must be completed. All tests must be approved without any errors before a license can be issued. The test course is agreed upon individually between the license applicant and the DGA.

6.1.1 End-points for services in the test environment

The end-points for the services on the DGA's test environment can be found below. The end-points covers services for TamperToken and ROFUS including the function "No thank you to gambling commercials", which must be used to perform the tests of the two systems. For both systems the service calls consists of web-services reached through the internet.

If the applicant completes the technical and legal/economic part of the application process and obtains a license, the DGA will inform the license holder of the end-points to services used for production.

TamperToken services for the TEST-environment:
Without certifikat: http://rofusdemo.spillemyndigheden.dk/TamperTokenAnvend/TamperTokenAnvendService

With certifikat: https://rofusdemo.spillemyndigheden.dk/TamperTokenAnvend/TamperTokenAnvendService

ROFUS services for the TEST-environment:
Without certifikat: http://rofusdemo.spillemyndigheden.dk/GamblerProject/GamblerService

With certifikat: https://rofusdemo.spillemyndigheden.dk/GamblerProject/GamblerService

ROFUS – "No thank you to gambling commercials"
Without certifikat:
http://rofusdemo.spillemyndigheden.dk/GamblerReklameProject/GamblerReklameService

With certifikat:
https://rofusdemo.spillemyndigheden.dk/GamblerReklameProject/GamblerReklameService

6.1.2 License applicant's connectivity test

The license applicant can perform the connectivity test the following way:

1) End-point is opened in an internet browser the following way respectively for TamperToken and ROFUS:
   a. https://rofusdemo.spillemyndigheden.dk/TamperTokenAnvend/TamperTokenAnvendService
   b. https://rofusdemo.spillemyndigheden.dk/GamblerProject/GamblerService
2) A login screen is shown and the applicant can type in the credentials issued by the DGA
3) If the connectivity-test is successful, the WSDL-file will be shown in the browser.

6.2 Test evaluation

The DGA evaluates the applicant's test results by reviewing the test case document and the provided documentation (screen dumps/logfile).

If the test results are correct and the documentation is sufficient, the applicant's test of TamperToken and ROFUS is approved.

If the test results are not correct and/or the documentation is not sufficient, the DGA will notify the applicant with the reason for not approving the test. This starts a process where the applicant must improve their test results and report this to the DGA for a new evaluation of the test.

When the applicant's tests are approved, the DGA performs a final review. The DGA reserves the right to demand further testing or documentation.

The applicant will receive a message when the review of the applicants test of technical requirements have been finalized.

In connection with the issue of the license, the DGA will create an account for the TamperToken and ROFUS production environment and the applicant will receive the credentials and end-points to production environment.

6.3 Access to the test environment after a license is issued

The DGA does not allow a general ongoing access to the test environment. Access can be granted upon request, if there is a specific need.

If a license holder needs access to the TamperToken or ROFUS test environment the DGA must be contacted. The license holder must give the following information:

- The license holder's username for the test environment (the same username, which was used in the application process)
- Information about what the license holder needs to test
- For how long the license holder expects the test to last.

In this connection the license holder can receive the test cases used during the application process. The ROFUS test case contains CPR-numbers, which can be used for testing of ROFUS functionalities.

7. Addition or change of gambling system

A license holder must inform the DGA in situations where the license holder wants to add an extra gambling system to their offer or move their existing offer of games entirely or partially from one gambling system to a new gambling system.

The situation described above is comparable to the process the DGA goes through when handling the technical part of a new application for a license, because these cases concerns new gambling systems with compositions, which the DGA does not have any prior knowledge about.

The license holder must fill in a new Annex B with associated documentation including certification reports. Furthermore, the DGA will require completion of a test case regarding ROFUS. A TamperToken test case must also be completed if the license holder changes the SAFE cf. section 3.8. In addition, the license holder must report new test data, so the DGA can verify, that the new gambling system reports correct game data.

8 License holder's obligations to notify

8.1 New games and changes in existing offer of games

This section contains a description of situations regarding changes in their offer of games, where the license holder is obligated to notify the DGA.

The requirements are also described in section 6 in the "Change Management Programme", which is part of the DGAs certification programme.

8.1.1 Implementation of new games

The implementation of new games, which does not affect how the license holder utilises the DGA's Standard Records, can commence without prior notification with the DGA.

The offering of new games, which utilises the DGA's Standard Records not previously utilised by the license holder, shall be notified with the DGA five working days before the offering commences and examples of Standard Records must be submitted along with the notification.

8.1.2 Changes in the existing offer of games

Changes to the existing offering of games, which does not affect how the license holder utilises the DGA's standard records, can commence without prior notification with the DGA.

Changes to the existing offering of games, which would affect the utilisation of the DGA's existing standard records by the license holder, shall be notified with the DGA five working days before the offering is changed and examples of Standard Records must be submitted along with the notification.

8.1.3 Situations where the DGA's standard records cannot be utilised

The offering of new games, which cannot utilise the DGA's Standard Records, shall be notified with the DGA at least 60 working days before the offering commences and shall not commence without prior approval from the DGA.

Changes to the existing offering of games, which would affect the utilisation of the DGA's Standard Records to an extent where they can no longer be used by the license holder, shall be notified with the DGA at least 60 days before the offering is changed and shall not commence without prior approval from the DGA.

8.2 Other obligations to notify

According to section G in Annex 1 to the Executive Order on online casino, land based betting and online betting the license holder must notify the DGA immediately when errors or violations occur, or when suspicion of errors or violations occur. The obligation to notify covers situations committed by either the license holder and/or business partners e.g. game suppliers.

This means that when an error occurs in a game, which is offered by the license holder, the DGA must be notified.

Furthermore, the license holder is obligated to notify the DGA when the conditions, on which the license is issued, changes significantly.

Concerning the technical requirements the obligation to notify covers situations, where the information given in Annex B to the application changes. This means that the license holder must notify the DGA, when the license holder engages a new game supplier, changes the registration and/or login process or moves the gambling system to a new physical location etc.

If the license holder makes changes to their SAFE, the DGA must also be notified cf. section 3.8 in this document.

Appendix 1

In connection with the application for a license, or when a license holder changes gambling system the DGA requires reporting of test data. All test data must be reported using TamperToken, be placed on a SAFE and reported to the DGA's test environment. For each game type it is described, which and how many standard records must be reported:

- Betting pool games: At least one total pool game.
  - PuljespilStartStruktur (one file pr. pool game)
  - PuljespilTransaktionStruktur (normally several files pr. pool game)
  - PuljespilEndOfGameStruktur (one file pr. pool game)
  - PuljespilSlutStruktur (one file pr. pool game)
- Poker tournament: At least one total poker tournament.
  - PokerTurneringStartStruktur (one file pr. poker tournament)
  - PokerTurneringTransaktionStruktur (normally several files pr. poker tournament)
  - PokerTurneringSlutStruktur (one file pr. poker tournament)
- Poker cash games: At least 20 sessions.
  - PokerCashGamePrSessionStruktur (can be reported in one xml file)
- Manager games: At least one total manager game.
  - ManagerspilStartStruktur (one file pr. manager game)
  - ManagerspilTransaktionStruktur (normally several files pr. manager game)
  - ManagerspilSlutStruktur (one file pr. manager game)
- Fixed odds betting: At least 50 fixed odds bets covering both the bet and the settlement.
  - FastOddsTransaktionStruktur (normally several bets pr. xml file)
  - FastOddsSlutStruktur (normally several settlements pr. xml file)
- Betexchange: At least 50 examples of bets on betexchange.
  - FastOddsTransaktionStruktur (normally several bets pr. xml file)
  - FastOddsSlutStruktur (normally several settlements pr. xml file)
- KasinoSinglePlayer: At least 20 sessions.
  - KasinoPrSessionStruktur (can be reported in one xml file)
- KasinoMultiPlayer: At least 20 sessions.
  - KasinoPrSessionStruktur (can be reported in one xml file)
- End Of Day: At least one day of mixed data with corresponding End Of Day reports.
  - Mixed standard records from a typical day. (all relevant standard records – how many of each standard record can be seen above)
  - EndOfDayRapportStruktur (One report for each game category for each currency played by Danish players during the day)

When reporting test data, the applicant or license holder must cover all scenarios offered from their gambling system. This means that if more types of casino games are offered, e.g. roulette, black jack and slot machines, then test data must cover all three types of casino games. This also means that if the applicant or license holder offers game on both desktop computers, smart phones and land based, then test data must cover all three sales channels.

Furthermore, applicants must demonstrate that they can handle the procedure for cancellations of transactions, and reporting of replacement data. This means, that the applicant

must cancel at least one of the reported test transactions, e.g. a fixed odds bet or a casino session, and replace e.g. an End Of Day report.