scytl

Secure Electronic Voting

RBAC on e-Voting

(and reporting)

Version 1.1

Date 16/06/2011

DOCUMENT HISTORY

Version	Date	Author	Comments
0.9	30/03/2011	Scytl R&D	First full version
1.0	04/05/2011	Scytl R&D	Updated version with some capital letters corrections, and including a roles description.
1.1	16/06/200	Scytl R&D	Copyright and disclaimer has been included. RBAC strings for uploading keys on KS have been specified.

DISCLAIMER: Some information in it this document might be obsolete, inaccurate or might be missing. Updates will be made if such discrepancies are found. This disclaimer will be also updated to reflect the state of the document.

NEXT VERSION TARGET DATE:

Version	Date	Author	Comments

QUALITY ASSURANCE:

Version	Date	QA responsible	Comments

Table of Contents

1. Methodology

In order to define – not only the functionalities which need to be protected through RBAC tokens – all the securable objects and their related permissions, in all servers and situations where the information is being managed, the following methodology has been applied.

1) Defining the list of objects which must be protected (securable objects). E.g. the election configuration files, the ballot box, the voting receipt list…

2) Create a generic list of actions which could be theoretically performed over the objects (securable actions over the securable objects).

   This generic list contains the following actions:

   - **Import-upload**.
   - **Update / Insert**.
   - **Delete**.
   - **Read - Query**.
   - **Export**.

   This list of actions is generic for all the objects, but there are exceptions where other actions can be analyzed.

3) Indicating the circumstances where the securable actions can be granted over the securable objects in a specific component. The following situations have been considered:
   - **nobody**. This action cannot be allowed nor implemented by the application (or directly it has no sense). E.g. it is not possible to import the ballot box in the VCS (object = Ballot box, action = import, component = VCS), or it is not possible to update the ballot box in the cleansing service (object = ballot box, action = update, component = cleansing).
   - **RBAC**. This action shall request for an specific RBAC token.E.g. to export the Ballot Box from the VCS you need to be specifically granted (object = Ballot Box, action = export, component = VCS).
   - **App**. This is an automatic action performed by the application, not by a user requests. E.g. the voting receipt list is updated by the RCG on each vote casting (object = Voting receipt list, action = update, component = RCG).

- **Everybody**. There is not any condition to execute the action, e.g. everybody is able to export the applet (object = applet, action = export, component = AS), or everybody is able to export the audit information from the Mixing server (object = Mixing Process, action = export tests, component = Mixing).

4) There are two results of this work:
    a) Filtering only by RBAC, a list of RBAC-strings related to the securable objects is generated to be implemented in the software.
    b) It can be useful to review all the accesses to the securable objects, not only when RBAC is involved, but also when the application is accessing to the objects.

2. RBAC and Securable objects (e-Voting)

2.1. Securable Objects

The following securable objects and location relationships have been defined:

Securable Object	Locations
Applet	AS
KeyStore	ALL servers
Election Configuration Files	VCS, RCG, Cleansing, Mixing, Counting
Election Configuration Database	VCS,RCG
Voter Credentials (P12)	AS
Authenticated Tokens List	AS
Return Codes	RCG
Voting Receipts	RCG, Cleansing
Ballot Box	VCS, Cleansing
Mark-offs	Cleansing
Electoral Roll	Cleansing
Cleansed Ballot Box	Cleansing, Mixing
Mixing Process	Mixing
Mixed Ballot Box	Mixing, Counting
Decrypted Ballot Box	Counting
Counts	Counting

2.2. Actions

The following table is the analysis of the possible actions over the identified securable objects per location.

Location	Securable Object	Actions	Access (RBAC/App/nobody)	String for RBAC	Comments
AS	Applet	Import-upload	nobody	-	(manual procedure)
AS	Applet	Update / Insert	nobody	-	
AS	Applet	Delete	nobody	-	
AS	Applet	Read - Query	everybody	-	Applet is public
AS	Applet	Export	everybody	-	Applet is public
AS	Voter Credentials (P12)	Import-upload	RBAC	e.AS.votercredentials.upload	
AS	Voter Credentials (P12)	Update / Insert	nobody	-	
AS	Voter Credentials (P12)	Delete	RBAC	e.AS.deleteelectionevent	Delete Election Event
AS	Voter Credentials (P12)	Read - Query	App	-	
AS	Voter Credentials (P12)	Export	App	-	
ALL	Election Configuration Files	Import-upload	RBAC	e.COMPONENT.emlfile.sendEML e.RCG.PublicParams.upload e.Cleansing.Areas.import e.Counting.Areas.import	
VCS, RCG, Cleansing, Mixing, Counting	Election Configuration Files	Update / Insert	nobody	-	
VCS/RCG	Election Configuration Files	Delete	RBAC	e.COMPONENT.deleteelectionevent	Delete Election Event NOT FOR AIR-GAP SERVERS
CL/MX/CO	Election Configuration Files	Delete	nobody	-	
ALL	Election Configuration Files	Read - Query	App	-	

Location	Securable Object	Actions	Access (RBAC/App/nobody)	String for RBAC	Comments
VCS	Election Configuration Files	Export	everybody	-	Ballot template on the VCS
RCG/CL/ MX/CO	Election Configuration Files	Export	nobody	-	
VCS/RCG	Election Configuration Database	Import-upload	App	-	
VCS/RCG	Election Configuration Database	Update / Insert	nobody	-	
VCS/RCG	Election Configuration Database	Delete	RBAC	e.COMPONENT.deleteelectionevent	Delete Election Event
VCS/RCG	Election Configuration Database	Read - Query	App	-	
VCS/RCG	Election Configuration Database	Export	nobody	-	
ALL	KeyStore	Import-upload	nobody	-	The import action is only executed to import the RBAC certificate; so it is not possibleto verify the RBAC token for this.
ALL	KeyStore	Update / Insert	RBAC	e.COMPONENT.KeyStore.sendkey e.COMPONENT.KeyStore.sendfile	There are two actions, one for sending the key to the keystore and other for sending the password to the component.
ALL	KeyStore	Delete	RBAC	e.COMPONENT.deleteelectionevent	Delete Election Event
ALL	KeyStore	Read - Query	App	-	
ALL	KeyStore	Export	nobody	-	
AS	Authenticated Tokens List	Import-upload	nobody	-	
AS	Authenticated Tokens List	Update / Insert	App	-	The AS keeps a list of authenticated tokens which have been used, to ensure its uniquesness.
AS	Authenticated Tokens List	Delete	nobody	-	
AS	Authenticated Tokens List	Read - Query	App	-	
AS	Authenticated Tokens List	Export	nobody	-	

Location	Securable Object	Actions	Access (RBAC/App/nobody)	String for RBAC	Comments
RCG	Return Codes	Import-upload	RBAC	e.RCG.ReturnCodes.upload	
RCG	Return Codes	Update / Insert	nobody	-	
RCG	Return Codes	Delete	RBAC	e.RCG.deleteelectionevent	Delete Election Event
RCG	Return Codes	Read - Query	App	-	
RCG	Return Codes	Export	nobody	-	
RCG	Voting Receipts	Import-upload	nobody	-	
RCG	Voting Receipts	Update / Insert	App	-	
RCG	Voting Receipts	Delete	RBAC	e.RCG.deleteelectionevent	Delete Election Event
RCG	Voting Receipts	Read - Query	App	-	
RCG	Voting Receipts	Export	RBAC	e.RCG.VotingReceipts.export	
VCS	Ballot Box	Import-upload	nobody	-	
VCS	Ballot Box	Update / Insert	App	-	
VCS	Ballot Box	Delete	RBAC	e.VCS.deleteelectionevent	Delete Election Event
VCS	Ballot Box	Read - Query	App	-	
VCS	Ballot Box	Export	RBAC	e.VCS.BallotBox.export	
Cleansing	Ballot Box	Import-upload	RBAC	e.Cleansing.BallotBox.import	
Cleansing	Ballot Box	Update / Insert	nobody	-	
Cleansing	Ballot Box	Delete	nobody	-	
Cleansing	Ballot Box	Read - Query	App	-	
Cleansing	Ballot Box	Export	nobody	-	
Cleansing	Voting Receipts	Import-upload	RBAC	e.Cleansing.VotingReceipts.import	
Cleansing	Voting Receipts	Update / Insert	nobody	-	
Cleansing	Voting Receipts	Delete	nobody	-	
Cleansing	Voting Receipts	Read - Query	App	-	
Cleansing	Voting Receipts	Export	nobody	-	
Cleansing	Mark-offs	Import-upload	RBAC	e.Cleansing.markoffs.import	ONLY PAPER MARK-OFFS

Location	Securable Object	Actions	Access (RBAC/App/nobody)	String for RBAC	Comments
Cleansing	Mark-offs	Update / Insert	App	-	ONLY e-Voting MARK-OFFS
Cleansing	Mark-offs	Delete	nobody	-	
Cleansing	Mark-offs	Read - Query	App	-	
Cleansing	Mark-offs	Export	RBAC	e.Cleansing.markoffs.export	ONLY e-Voting MARK-OFFS
Cleansing	Electoral Roll	Import-upload	RBAC	e.Cleansing.electoralroll.import	
Cleansing	Electoral Roll	Update / Insert	nobody	-	
Cleansing	Electoral Roll	Delete	nobody	-	
Cleansing	Electoral Roll	Read - Query	App	-	
Cleansing	Electoral Roll	Export	nobody	-	
Cleansing	Cleansed Ballot Box	Import-upload	nobody	-	
Cleansing	Cleansed Ballot Box	Update / Insert	App	-	
Cleansing	Cleansed Ballot Box	Delete	nobody	-	
Cleansing	Cleansed Ballot Box	Read - Query	App	-	
Cleansing	Cleansed Ballot Box	Export	RBAC	e.Cleansing.CleansedBB.export	
Mixing	Cleansed Ballot Box	Import-upload	RBAC	e.Mixing.CleansedBB.import	
Mixing	Cleansed Ballot Box	Update / Insert	nobody	-	
Mixing	Cleansed Ballot Box	Delete	nobody	-	
Mixing	Cleansed Ballot Box	Read - Query	App	-	
Mixing	Cleansed Ballot Box	Export	nobody	-	
Mixing	Mixing Process	Execute	RBAC	e.Mixing.mixing.execute	
Mixing	Mixing Process	Enter Random	RBAC	e.Mixing.audit.random	
Mixing	Mixing Process	Export tests	everybody	-	
Mixing	Mixing Process	Validate Tests	RBAC	e.Mixing.audit.validate	
Mixing	Mixed Ballot Box	Import-upload	nobody	-	
Mixing	Mixed Ballot Box	Update / Insert	App	-	
Mixing	Mixed Ballot Box	Delete	nobody	-	

Location	Securable Object	Actions	Access (RBAC/App/nobody)	String for RBAC	Comments
Mixing	Mixed Ballot Box	Read - Query	nobody	-	
Mixing	Mixed Ballot Box	Export	RBAC	e.Mixing.MixedBB.export	
Counting	Mixed Ballot Box	Import-upload	RBAC	e.Counting.MixedBB.import	
Counting	Mixed Ballot Box	Update / Insert	nobody	-	
Counting	Mixed Ballot Box	Delete	nobody	-	
Counting	Mixed Ballot Box	Read - Query	App	-	
Counting	Mixed Ballot Box	Export	nobody	-	
Counting	Decrypted Ballot Box	Import-upload	nobody	-	
Counting	Decrypted Ballot Box	Update / Insert	RBAC	e.Counting.decrypt	
Counting	Decrypted Ballot Box	Delete	nobody	-	
Counting	Decrypted Ballot Box	Read - Query	App	-	
Counting	Decrypted Ballot Box	Export	nobody	-	
Counting	Counts	Import-upload	nobody	-	
Counting	Counts	Update / Insert	App	-	
Counting	Counts	Delete	nobody	-	
Counting	Counts	Read - Query	everybody	-	
Counting	Counts	Export	RBAC	e.Counting.counts.export	

2.3. RBAC strings

According to the securable objects analysis, the following are the RBAC strings implemented in the e-voting application, which need to be assigned to user roles.

RBAC strings	RBAC description
e.AS.votercredentials.upload	Upload voter credentials in the Authentication Service.
e.VCS.emlfile.sendEML	Import election configuration in the VCS.
e.RCG.emlfile.sendEML	Import election configuration in the RCG.
e.Cleansing.emlfile.sendEML	Import election configuration in the Cleansing.
e.Mixing.emlfile.sendEML	Import election configuration in the Mixing.
e.Counting.emlfile.sendEML	Import election configuration in the Counting.
e.RCG.PublicParams.upload	Import election configuration in the RCG (public params).
e.Cleansing.Areas.import	Import election configuration in the Cleansing (areas).
e.Counting.Areas.import	Import election configuration in the Counting (areas).
e.AS.KeyStore.sendkey e.AS.KeyStore.sendfile	Upload a cryptographic key to the AS.
e.VCS.KeyStore.sendkey e.VCS.KeyStore.sendfile	Upload a cryptographic key to the VCS.
e.RCG.KeyStore.sendkey e.RCG.KeyStore.sendfile	Upload a cryptographic key to the RCG.
e.TPM.KeyStore.sendkey e.TPM.KeyStore.sendfile	Upload a cryptographic key to the TPM.
e.Cleansing.KeyStore.sendkey e.Cleansing.KeyStore.sendfile	Upload a cryptographic key to the Cleansing.
e.Mixing.KeyStore.sendkey e.Mixing.KeyStore.sendfile	Upload a cryptographic key to the Mixing.
e.Counting.KeyStore.sendkey e.Counting.KeyStore.sendfile	Upload a cryptographic key to the Counting.
e.RCG.ReturnCodes.upload	Upload the generated Return Codes in the RCG.
e.RCG.VotingReceipts.export	Export the Voting Receipts from the RCG.
e.VCS.BallotBox.export	Export the Ballot Box from the VCS.
e.Cleansing.BallotBox.import	Import the Ballot Box in the Cleansing.
e.Cleansing.VotingReceipts.import	Import the Voting Receipts in the Cleansing.
e.Cleansing.markoffs.import	Import the list of paper-voters (paper markoffs) in the Cleansing.
e.Cleansing.markoffs.export	Export the list of e-voters (e-voters markoffs) from the Cleansing.
e.Cleansing.electoralroll.import	Import the Electoral Roll information in the Cleansing Service.
e.Cleansing.CleansedBB.export	Export the cleansed Ballot Box from the Cleansing.

RBAC strings	RBAC description
e.Mixing.CleansedBB.import	Import the cleansed Ballot Box in the Mixing.
e.Mixing.mixing.execute	Execute the Mixing process.
e.Mixing.audit.random	Introduce a random value to perform the Mixing process.
e.Mixing.audit.validate	Introduce Auditor results to the mixing tests.
e.Mixing.MixedBB.export	Export the mixed Ballot Box from the Mixing.
e.Counting.MixedBB.import	Import the mixed Ballot Box in the Counting Server.
e.Counting.decrypt	Decrypt the votes in the Counting Server.
e.Counting.counts.export	Export the counting results from the Counting Server.
e.AS.deleteelectionevent e.VCS.deleteelectionevent e.RCG.deleteelectionevent	Delete all information from an election event (not air-gap servers).

2.4. Proposed list of roles on e-Voting

The following is a PROPOSAL of roles which can be defined and managed in the Admin application, regarding e-Voting functionalities.

Role	Description	Strings
Election Officials	Election configuration (online servers)	e.AS.votercredentials.upload e.VCS.emlfile.sendEML e.RCG.emlfile.sendEML e.RCG.PublicParams.upload e.RCG.ReturnCodes.upload e.AS.KeyStore.sendkey e.VCS.KeyStore.sendkey e.RCG.KeyStore.sendkey e.TPM.KeyStore.sendkey e.KS.KeyStore.sendkey e.AS.KeyStore.sendfile e.VCS.KeyStore.sendfile e.RCG.KeyStore.sendfile e.TPM.KeyStore.sendfile e.KS.KeyStore.sendfile
	Election configuration (air-gap servers)	e.Cleansing.emlfile.sendEML e.Mixing.emlfile.sendEML e.Counting.emlfile.sendEML e.Cleansing.Areas.import e.Counting.Areas.import e. Cleansing.KeyStore.sendkey e. Mixing.KeyStore.sendkey e. Counting.KeyStore.sendkey e.Cleansing.KeyStore.sendfile e.Mixing.KeyStore.sendfile e.Counting.KeyStore.sendfile
	Tallying Operations (online servers)	e.VCS.BallotBox.export e.RCG.VotingReceipts.export

Role	Description	Strings
	Tallying Operations (air-gap servers)	e.Cleansing.BallotBox.import e.Cleansing.VotingReceipts.import e.Cleansing.markoffs.import e.Cleansing.markoffs.export e.Cleansing.electoralroll.import e.Cleansing.CleansedBB.export e.Mixing.CleansedBB.import e.Mixing.mixing.execute e.Mixing.MixedBB.export e.Counting.MixedBB.import e.Counting.counts.export
Mixing Auditor	Mixing auditing	e.Mixing.mixing.execute e.Mixing.audit.random e.Mixing.audit.validate
Electoral Board member	Decrypting	e.Counting.decrypt
	Delete Election Event	e.AS.deleteelectionevent e.VCS.deleteelectionevent e.RCG.deleteelectionevent

3. RBAC in the REPORTING system

3.1. RBAC strings

At the time an admin application system user is created, a set of reporting permissions is assigned to him, allowing the user to operate on templates and reports. These permissions are:

Permission	RBAC string
Download Kit	*e.reporting.kit.download*
Upload Template	*e.reporting.template.upload*
Download Template	*e.reporting.template.download*
Edit Template	*e.reporting.template.edit*
Duplicate Template	*e.reporting.template.duplicate*
Validate Template	*e.reporting.template.validate*
Deactivate Template	*e.reporting.template.deactivate*
Reactivate Template	*e.reporting.template.reactivate*
Delete Template	*e.reporting.template.delete*
Execute template	*e.reporting.template.execute*
Execute (In process)	*e.reporting.template.execute_in_process*
Download Report	*e.reporting.report.download*
Validate Report	*e.reporting.report.validate*
Deactivate Report	*e.reporting.report.deactivate*
Reactivate Report	*e.reporting.report.reactivate*
Delete Report	*e.reporting.report.delete*

In addition to this RBAC strings, the following ones – that by hierarchy are superior, and are granting access to all the inferior levels – exist and are not recommended to be assigned to any role:

- **e.reporting**: It means having full access to reporting.
- **e.reporting.template**: It means having full access to templates (create, validate, delete…).
- **e.reporting.report**: It means having full access to reports (create, validate, delete…).

3.2. Proposed list of roles on reporting

The following is a PROPOSAL of roles which can be defined and managed in the Admin application, regarding the reporting functionalities.

Role	Description	Strings
Reporting Creator	This role is allowed to create and edit templates.	*e.reporting.kit.download* *e.reporting.template.download* *e.reporting.template.upload* *e.reporting.template.edit* *e.reporting.template.duplicate* *e.reporting.template.delete*
Reporting Validator	The reporting validator role shall validate templates and reports before its acceptance.	*e.reporting.template.validate* *e.reporting.template.deactivate* *e.reporting.template.reactivate* *e.reporting.template.delete* *e.reporting.report.deactivate* *e.reporting.report.reactivate* *e.reporting.report.validate* *e.reporting.report.delete*
Reporting Executor	This role is able to execute a template for generating a report. It has also access to download the report, and delete it (to allow an executor review his work).	*e.reporting.template.execute* *e.reporting.template.execute_in_process* *e.reporting.report.download* *e.reporting.report.delete*
Reporting Viewer	This role is only allowed to download a generated report.	*e.reporting.report.download*