

Public Verification System (PubVeriS) for eVoting Version 1.0
Interface Definition
Alexander Prosser

Version 1.0, August 2009

Terms of Usage:

This definition describes the data interface for providing ballot box data for analysis in PubVeriS. Software implementers may use the interface description solely for the purpose to write software that provides data to be checked with PubVeriS. Any other usage is prohibited.

Usage of the interface definition by a software implementer does not imply any co-authorship or shared copyright with the author or copyright holder of this interface definition. It is the software implementer's sole responsibility to provide the appropriate data consistent with the processes in his system.

For further terms, see the license agreement.

1 Interface

1.1 Files

The interface consists of the following files:

keys.eml
options.eml
ballotbox.eml

1.2 keys.eml

Consists of the following key structures:

```
<VoteEncryptionKey>  
<e>...</e>  
<d>...</d>  
<m>...</m>  
</VoteEncryptionKey>
```

This key was used to encrypt the votes and may now (including the private key d) be used to decrypt them. Keys are in lower case, hexadecimal format.

```
<BlindSignatureKey>  
<e>...</e>  
<m>...</m>  
</BlindSignatuerKey>
```

This key was used to blindly sign the tokens and may now be used in its public part to verify the signature. Key format as above.

If a verifier was used, the following blind signature key for the verifier is to be expected as well:

```
<BlindSignatureKeyVF>  
<e>...</e>  
<m>...</m>  
</BlindSignatureKeyVF>
```

1.3 options.eml

```

<Election>
  <ElectionIdentifier> numeric </ElectionIdentifier>
  <ElectionName> string </ElectionName>
  <Verifier> {"False" | "True"} </Verifier>
  <Contest>
    <ContestIdentifier> numeric </ContestIdentifier>
    <NumberOfVotes> numeric </NumberOfVotes>
    <NumberOfCumulVotes> numeric </NumberOfCumulVotes>
    <NumberOfStrikeouts> numeric </NumberOfStrikeouts>
    <NumberOfSuboptions> numeric </NumberOfSuboptions>
    <NumberOfCumulSuboptions>
      numeric
    </NumberOfCumulSuboptions>
    <NumberOfStrikeoutsSuboptions>
      numeric
    </NumberOfStrikeoutsSuboptions>
    <Panager> {"False" | "True"} </Panager>
    <Validation> {"False" | "True"} </Validation>
    <Options>
      <Selection>
        <AffiliationIdentifier>
          numeric
        </AffiliationIdentifier>
        <AffiliationName> string </AffiliationName>
        <invalid> {0 | 1 | 2} </invalid>
      </Selection>
      <Selection>
        <AffiliationIdentifier>
          numeric
        </AffiliationIdentifier>
        <AffiliationName> string </AffiliationName>
        <invalid> {0 | 1 | 2} </invalid>
      </Selection>
      ...
      ...
    </Options>
    <SubOptions>
      <Selection>
        <CandidateIdentifier>
          numeric
        </CandidateIdentifier>
        <CandidateName> string </CandidateName>
        <AffiliationIdentifier>
          numeric
        </AffiliationIdentifier>
      </Selection>
      <Selection>
        <CandidateIdentifier>
          numeric
        </CandidateIdentifier>
        <CandidateName> string </CandidateName>
        <AffiliationIdentifier>
          numeric

```

```

        </AffiliationIdentifier>
    </Selection>

    </SubOptions>
</Contest>
</Election>

```

The fields following the <Contest> tag designate the constituency. Note that only one constituency is processed per download process. Parameters

```

    <NumberOfVotes> numeric </NumberOfVotes>
    <NumberOfCumulVotes> numeric </NumberOfCumulVotes>
    <NumberOfStrikeouts> numeric </NumberOfStrikeouts>
    <NumberOfSuboptions> numeric </NumberOfSuboptions>
    <NumberOfCumulSuboptions>
        numeric
    </NumberOfCumulSuboptions>
    <NumberOfStrikeoutsSuboptions>
        numeric
    </NumberOfStrikeoutsSuboptions>

```

indicate (in the above order) the number of permissible votes for main options, the number of votes that may be cumulated in a main options, the number of main options that may be struck out, the number of votes for sub options, the max. cumulation in a sub option and the number of sub options that may be struck out.

```

    <Panager> {"False" | "True"} </Panager>
    <Validation> {"False" | "True"} </Validation>

```

These parameters indicate whether splitting the ticket is possible (Panager) and whether the vote is validated in the ballot sheet presented to the voter.

The <Options> section lists the available main options. Each option is wrapped in a <Selection> tag.

The <SubOptions> section lists the available sub options ("candidates") and the main option they are linked to (tag <AffiliationIdentifier>). Also here, each individual sub options is wrapped in a <Selection> tag.

```

    <invalid> {0 | 1 | 2} </invalid>

```

This parameter indicates, whether the respective option is not "invalid", "invalid" or "invalid all".

<Selection> tags in main and sub options may occur an arbitrary number of times. The affiliation referenced in any sub option must exist in the <Options> section.

1.4 ballotbox.eml

```
<?eml version=5.0 encoding=utf-8 type=460?>
<CastVote>
  <VToken> string </VToken><Vote> string </Vote>
  ...
  ...
</CastVote>
```

Within <CastVote> there is one line per vote, each containing a <VToken> and a <Vote> element. In the downloaded state, vote contains the encrypted ballot in lower case hex string format. The line may occur an arbitrary number of times.

The internal structure of <VToken> is

```
<VTokenRequest> string </VTokenRequest>
<VTokenResponseServer> string </VTokenResponseServer>
<VTokenResponse> string </VTokenResponse>
```

All strings are lower case hex strings.

1.5 Decrypted Votes

<VoteEncryptionKey> from keys.eml is used to encrypt the content <vote> from ballotbox.eml. The encryption blocks are designated by a <Block> tag for robust processing.

Each <Vote> has the following structure after decryption:

```
<id> numeric </id>
<VTR> string </VTR>
<v>
  <i> numeric </i>
  <val> numeric </val>
</v>
...
<s>
  <i> numeric </i>
  <val> numeric </val>
</s>
...
```

A value of -1 in <val> indicates a strike-out. The following consistency criteria apply:

- ◆ <id> must be the same as <ContestIdentifier> in options.eml.
- ◆ <VTR> must be identical to <VTokenRequest> in the token.

Trademarks:

Java is a trademark of Sun Microsystems (www.sun.com).

The Election Markup Language EML is a standard provided by OASIS (www.oasis-open.org).

RSA is a trademark of RSA Systems, a division of EMC (www.rsa.com).

AES is a standard of NIST (www.nist.gov) as FIPS 197.