

API Reference

# ZofToken API

API Version: 1.0

# INDEX

|                            |           |
|----------------------------|-----------|
| <b>1. ENROLLMENT</b>       | <b>3</b>  |
| 1.1 POST /enrollment       | 3         |
| <b>2. IDENTCODES</b>       | <b>4</b>  |
| 2.1 GET /identcodes        | 4         |
| <b>3. OFFLINECHALLENGE</b> | <b>6</b>  |
| 3.1 GET /offlinechallenge  | 6         |
| <b>4. OFFLINERESPONSE</b>  | <b>7</b>  |
| 4.1 POST /offlinerresponse | 7         |
| <b>5. REMOVAL</b>          | <b>8</b>  |
| 5.1 POST /removal          | 8         |
| <b>6. STATISTICS</b>       | <b>9</b>  |
| 6.1 GET /statistics        | 9         |
| <b>7. STATUS</b>           | <b>10</b> |
| 7.1 GET /status            | 10        |
| <b>8. STATUSADMIN</b>      | <b>11</b> |
| 8.1 POST /statusadmin      | 11        |

# API

## 1. ENROLLMENT

### 1.1 POST /enrollment

Prepare/reset a ZofToken for a user/service combination and obtain enrollment information

#### REQUEST

REQUEST BODY - application/json

```
{
  id*           string  User identifier
  service*      string  Service identifier
  authKey*      string  Authorization string
  codeSize*     number  Length of the code needed for enrollment (must be 4, 8, 12, 16, 20, 24, 28 or 32)
  forceReset    boolean  Must be set to true to reset an existing ZofToken
}
```

#### RESPONSE

STATUS CODE - 200: Enrollment information

RESPONSE MODEL - application/json

```
{
  code    string  Code required for manual enrollment
  qrCode  string  QR code for automatic enrollment (provided as a Data URL)
}
```

STATUS CODE - 400: Request error

RESPONSE MODEL - application/json

```
{
  error string  Error message
}
```

STATUS CODE - 403: Authorization error

RESPONSE MODEL - application/json

```
{
  error string  Error message
}
```

STATUS CODE - 500: Internal server error

RESPONSE MODEL - application/json

```
{
  error string  Error message
}
```

## 2. IDENTCODES

### 2.1 GET /identcodes

Get normal and duress ident codes to compare with one provided by a user through any channel

#### REQUEST

##### QUERY PARAMETERS

| NAME     | TYPE   | DESCRIPTION          |
|----------|--------|----------------------|
| *id      | string | User identifier      |
| *service | string | Service identifier   |
| *authKey | string | Authorization string |

#### RESPONSE

STATUS CODE - 200: Ident codes

RESPONSE MODEL - application/json

```
{
  current      string  Ident code that corresponds to the current 30 second window to check against user input
  next         string  Ident code that corresponds to the next 30 second window to check against user input (for
                    potential clock differences)
  previous     string  Ident code that corresponds to the previous 30 second window to check against user input (for
                    potential clock differences)
  currentDuress string  Duress ident code that corresponds to the current 30 second window to check against user
                    input - only available if token has a duress key configured
  nextDuress   string  Duress ident code that corresponds to the next 30 second window to check against user input
                    (for potential clock differences) - only available if token has a duress key configured
  previousDuress string Duress ident code that corresponds to the previous 30 second window to check against user
                    input (for potential clock differences) - only available if token has a duress key configured
}
```

STATUS CODE - 400: Request error

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

STATUS CODE - 403: Authorization error

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

STATUS CODE - 404: User not found

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

STATUS CODE - 500: Internal server error

## RESPONSE MODEL - application/json

```
{  
  error string Error message  
}
```

---

# 3. OFFLINECHALLENGE

## 3.1 GET /offlinechallenge

Get a challenge for opening a ZofToken in offline mode for a user/service combination

### REQUEST

#### QUERY PARAMETERS

| NAME      | TYPE   | DESCRIPTION  |
|-----------|--------|--|
| *id       | string | User identifier  |
| *service  | string | Service identifier   |
| *authKey  | string | Authorization string   |
| *codeSize | number | Length of the code to be used in offline mode (must be 4, 8, 12, 16, 20, 24, 28 or 32) |

### RESPONSE

STATUS CODE - 200: Offline challenge

RESPONSE MODEL - application/json

```
{
  offlineChallenge string Challenge code to be sent to the user to open a ZofToken in offline mode
}
```

STATUS CODE - 400: Request error

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

STATUS CODE - 403: Authorization error

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

STATUS CODE - 404: User not found

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

STATUS CODE - 500: Internal server error

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

# 4. OFFLINERESPONSE

## 4.1 POST /offlinerresponse

Attempt to open a ZofToken in offline mode for a user/service combination

### REQUEST

REQUEST BODY - application/json

```
{
  id*      string  User identifier
  service* string  Service identifier
  authKey* string  Authorization string
  response* string  Response provided by the ZofToken user in offline mode
}
```

### RESPONSE

STATUS CODE - 200: Status information

RESPONSE MODEL - application/json

```
{
  status number 1 - Open, 3 - Authenticated as under duress
}
```

STATUS CODE - 400: Request error

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

STATUS CODE - 403: Authorization error

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

STATUS CODE - 404: User not found

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

STATUS CODE - 500: Internal server error

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

# 5. REMOVAL

## 5.1 POST /removal

### Remove a ZofToken for a user/service combination

#### REQUEST

##### REQUEST BODY - application/json

```
{  
  id*      string  User identifier  
  service* string  Service identifier  
  authKey* string  Authorization string  
}
```

#### RESPONSE

**STATUS CODE - 200:** Confirmation message

##### RESPONSE MODEL - application/json

```
{  
  message string  
}
```

**STATUS CODE - 400:** Request error

##### RESPONSE MODEL - application/json

```
{  
  error string  Error message  
}
```

**STATUS CODE - 403:** Authorization error

##### RESPONSE MODEL - application/json

```
{  
  error string  Error message  
}
```

**STATUS CODE - 404:** User not found

##### RESPONSE MODEL - application/json

```
{  
  error string  Error message  
}
```

**STATUS CODE - 500:** Internal server error

##### RESPONSE MODEL - application/json

```
{  
  error string  Error message  
}
```



# 6. STATISTICS

## 6.1 GET /statistics

Obtain performance statistics from the server

### REQUEST

No request parameters

### RESPONSE

**STATUS CODE - 200:** Server performance statistics

**RESPONSE MODEL - application/json**

```
{
  totalUsers      number  Number of ZofTokens currently in use (in all states)
  opsLastSecond  number  Number of ZofToken operations (open/close) processed by the server in the last second
  opsLastMinute  number  Number of ZofToken operations (open/close) processed by the server in the last minute
  opsAvgTiming   number  Average processing time (in milliseconds) for ZofToken operations (open/close)
}
```

**STATUS CODE - 403:** Authorization error

**RESPONSE MODEL - application/json**

```
{
  error string  Error message
}
```

---

# 7. STATUS

## 7.1 GET /status

Get the ZofToken status for a user/service combination

### REQUEST

#### QUERY PARAMETERS

| NAME     | TYPE   | DESCRIPTION          |
|----------|--------|----------------------|
| *id      | string | User identifier      |
| *service | string | Service identifier   |
| *authKey | string | Authorization string |

### RESPONSE

**STATUS CODE - 200:** Status information

**RESPONSE MODEL - application/json**

```
{
  status number 0 - Awaiting enrollment, 1 - Open, 2 - Closed, 3 - Last authenticated as under duress, 4 - Blocked
}
```

**STATUS CODE - 400:** Request error

**RESPONSE MODEL - application/json**

```
{
  error string Error message
}
```

**STATUS CODE - 403:** Authorization error

**RESPONSE MODEL - application/json**

```
{
  error string Error message
}
```

**STATUS CODE - 404:** User not found

**RESPONSE MODEL - application/json**

```
{
  error string Error message
}
```

**STATUS CODE - 500:** Internal server error

**RESPONSE MODEL - application/json**

```
{
  error string Error message
}
```

# 8. STATUSADMIN

## 8.1 POST /statusadmin

Force a ZofToken into a particular status for a user/service combination

### REQUEST

REQUEST BODY - application/json

```
{
  id*      string  User identifier
  service* string  Service identifier
  authKey* string  Authorization string
  status*  number  New ZofToken status (only Open/Closed/Blocked allowed)
}
```

### RESPONSE

STATUS CODE - 200: Confirmation message

RESPONSE MODEL - application/json

```
{
  message string
}
```

STATUS CODE - 400: Request error

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

STATUS CODE - 403: Authorization error

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

STATUS CODE - 404: User not found

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```

STATUS CODE - 500: Internal server error

RESPONSE MODEL - application/json

```
{
  error string Error message
}
```