
KAPITOLA 1 - PŘÍLOHA 2 – ATEAS API	2
1.1. PRINCIP KOMUNIKACE	2
1.2. ATEAS API PRO ADMINISTRAČNÍ SERVER	3
1.2.1. EXTERNÍ UDÁLOSTI	3
1.2.2. VIDEO STĚNA	3
1.2.3. REGISTRAČNÍ ZNAČKY	5
1.2.4. DATABÁZE OBLIČEJŮ	6
1.2.5. NOTIFIKACE UDÁLOSTÍ	13
1.2.6. NOTIFIKACE PŘIHLÁŠENÍ	15
1.3. ATEAS API PRO KAMEROVÝ SERVER	16
1.3.1. EXTERNÍ UDÁLOSTI	16
1.3.2. METADATA	17
1.4. PARAMETRICKÉ SPOUŠTĚNÍ APLIKACÍ	18
1.4.1. ADMINISTRAČNÍ SERVER	18
1.4.2. KAMEROVÝ SERVER	19

Kapitola 1 - Příloha 2 – ATEAS API

1.1. Princip komunikace

Je možné využít komunikační kanál založený na protokolu TCP anebo HTTP. Použití TCP protokolu je jednodušší a může být podporováno širším spektrem nejrůznějších zařízení. Přenos však není možné nijak zabezpečit anebo vyžadovat autentizaci. U protokolu HTTP je možné vynutit použití TLS protokolu pro maximální zabezpečení a také volitelně vyžadovat přihlašovací údaje.

Při použití tohoto API je administrační či kamerový server stranou naslouchající na příslušných síťových portech, externí zařízení je stranou vytvářející a udržující spojení.

Administrační server naslouchá na API portu 8504 (TCP) a na portu 80 (HTTP), který je výchozím HTTP portem administračního serveru a lze ho změnit v konfiguračním souboru. Kamerový server naslouchá na API portu 8505 (TCP) a na portu 8080 (HTTP), který výchozím HTTP portem kamerového serveru a lze ho změnit v konfiguračním souboru.

Protokol HTTP používá standardní HTTP návratové kódy. Zasílání dat se provádí metodou HTTP POST. Protokol HTTP také může vracet data ve formátu JSON.

Pomocí protokolu TCP mohou být zasílány nevyžádané informace, např. o událostech v systému. Pro tento účel je na úrovni protokolu HTTP použit protokol WebSocket. Informace mohou být ve formátu JSON nebo XML.

Kódování textu u protokolu TCP je nastavitelné při otevírání portu, u protokolu HTTP se využívá kódování UTF8.

TIP

Odkaz na příklad implementace pro protokol TCP a HTTP naleznete na domovské stránce Vašeho administračního serveru.

1.2. ATEAS API pro administrační server

1.2.1. Externí události

Příjem externí události

Protokol	API
TCP	[ATEAS EVENT (status) (objectid) (elementid)]
HTTP	POST /api-base/event DATA: status=(status)&objectid=(objectid)&elementid=(elementid)

Parametry:

Parametr	Hodnota	Výchozí	Význam
status	START, STOP		Zahájení či ukončení
objectid	1 – 10 000		Číslo objektu
elementid	1 – 99 999		Číslo prvku

Příklad:

Protokol	API
TCP	[ATEAS EVENT START 1 1]
HTTP	DATA: status=START&objectid=1&elementid=1

1.2.2. Video stěna

Přepnutí obsahu na video stěnu

Protokol	API
TCP	[ATEAS VIDEOWALL (monitor) (submonitor) (serverid) (deviceid) (wallid) (meta)]
HTTP	POST /api-base/videowall DATA: monitor=(monitor)&submonitor=(submonitor)&serverid=(serverid)&deviceid=(deviceid)&wallid=(wallid)&meta=(meta)

Parametry:

Parametr	Hodnota	Výchozí	Význam
monitor	1 – 192		Číslo monitoru
submonitor	0 – 16	0	Číslo submonitoru
serverid	1 – 9 999		Číslo serveru
deviceid	1 – 999		Číslo zařízení
wallid	1 – 1000	1	Číslo video stěny
meta	0 – 1	0	Připojení metadat

Příklad:

Protokol	API
TCP	[ATEAS VIDEOWALL 1 1 1 10 1 0]
HTTP	DATA: monitor=1&submonitor=1&serverid=1&deviceid=10&wallid=1&meta=0

POZNÁMKA

Hodnotu submonitor je nutné použít pro monitory typu 4, 9 nebo 16.

POZNÁMKA

Jsou-li obě hodnoty serverid a deviceid rovny nule, dojde k vypnutí příslušného monitoru (vypnutí videa a přechod do výchozí podoby s logem ATEAS).

POZNÁMKA

Pokud je serverid nastaveno na hodnotu nula, je pozitivní deviceid interpretováno jako číslo URL webového obsahu vložené do systému administrátorem.

POZNÁMKA

U protokolu TCP je při použití nepovinného parametru wallid nutné použít i parametr submonitor, aby bylo možné zprávu správně zpracovat.

1.2.3. Registrační značky

Změna zařazení RZ

Protokol	API
TCP	
HTTP	POST /api-base/plate DATA: plate=(plate)&list=(list)

Parametry:

Parametr	Hodnota	Výchozí	Význam
plate	(značka)		Registrační značka
list	none, white, black, user1, user2		Zařazení do seznamu

Příklad:

Protokol	API
TCP	
HTTP	DATA: plate=2A56217&plate=white

JSON odpověď:

```
{
  "Plate": "2A56217",
  "List": "white"
}
```

POZNÁMKA

Registrační značky lze zadat v dekorativním či normalizovaném tvaru.

POZNÁMKA

Parametr list je nepovinný a pokud je vynechán, bude pouze zjištěno aktuální zařazení RZ do seznamu.

1.2.4. Databáze obličejů

Přidání skupiny osob

Protokol	API
TCP	
HTTP	POST /api-base/face/group/add DATA: name=(name)

Parametry:

Parametr	Hodnota	Výchozí	Význam
name	(name)		Název skupiny

Příklad:

Protokol	API
----------	-----

TCP	
HTTP	DATA: name=group

JSON odpověď:

```
{
  "result": "ok",
  "id": "2"
  "name": "group"
}
```

POZNÁMKA

Nelze vytvářet skupiny s duplicitním názvem bez ohledu na velká a malá písmena.

Odebrání skupiny osob

Protokol	API
TCP	
HTTP	POST /api-base/face/group/remove DATA: id=(id)

Parametry:

Parametr	Hodnota	Výchozí	Význam
id	(id)		Číslo skupiny

Příklad:

Protokol	API
TCP	

HTTP	DATA: id=2
-------------	------------

JSON odpověď:

```
{
  "result": "ok"
}
```

Přejmenování skupiny osob

Protokol	API
TCP	
HTTP	POST /api-base/face/group/name DATA: id=(id)&name=(name)

Parametry:

Parametr	Hodnota	Výchozí	Význam
id	(id)		Číslo skupiny
name	(name)		Nový název skupiny

Příklad:

Protokol	API
TCP	
HTTP	DATA: id=2&name=group

JSON odpověď:

```
{
  "result": "ok"
}
```



```
}

```

Přidání osoby

Protokol	API
TCP	
HTTP	POST /api-base/face/person/add DATA: name=(name)&group=(group)&uuid=(uuid)

Parametry:

Parametr	Hodnota	Výchozí	Význam
name	(name)		Jméno osoby
group	(id)	1	Číslo skupiny
uuid	(uuid)		Vlastní identifikátor

Příklad:

Protokol	API
TCP	
HTTP	DATA: name=person&group=&uuid=

JSON odpověď:

```
{
  "result": "ok",
  "id": "1",
  "name": "person",
  "uuid": ""
}
```

POZNÁMKA

Parametr uuid může sloužit jako externí identifikátor např. v přístupovém systému. Jeho hodnota je poté součástí události rozpoznání obličeje.

Odebrání osoby

Protokol	API
TCP	
HTTP	POST /api-base/face/person/remove DATA: id=(id)

Parametry:

Parametr	Hodnota	Výchozí	Význam
id	(id)		Číslo osoby

Příklad:

Protokol	API
TCP	
HTTP	DATA: id=1

JSON odpověď:

```
{
  "result": "ok"
}
```

Změna osoby

Protokol	API
TCP	
HTTP	POST /api-base/face/person/update DATA: id=(id)&name=(name)&group=(group)&uuid=(uuid)

Parametry:

Parametr	Hodnota	Výchozí	Význam
id	(id)		Číslo osoby
name	(name)		Nové jméno osoby
group	(id)		Číslo skupiny
uuid	(uuid)		Vlastní identifikátor

Příklad:

Protokol	API
TCP	
HTTP	DATA: id=1&name=person&group=2&uuid=

JSON odpověď:

```
{
  "result": "ok",
  "name": "person",
  "uuid": ""
}
```

Přidání obličeje

Protokol	API
----------	-----

TCP	
HTTP	POST /api-base/face/person/image DATA: person=(person)&serverid=(serverid)&index=(index)&data=(data)

Parametry:

Parametr	Hodnota	Výchozí	Význam
person	(id)		Číslo osoby
serverid	(id)		Číslo serveru pro analýzu
index	0 – 9	0	Index obrázku
data	(uuid)		Data obrázku

Příklad:

Protokol	API
TCP	
HTTP	DATA: person=1&serverid=1&index=0&data=

JSON odpověď:

```
{
  "result": "ok",
  "image": "/9j/4AA..."
}
```

POZNÁMKA

V parametru data je očekáván obrázek ve formátu jpeg či bmp v kódování base64 s barevnou hloubkou 24 nebo 32 bitů. Pokud je parametr data prázdný, dojde k odstranění obrázku.

POZNÁMKA

Po vložení obličeje do databáze je vrácen obrázek v kanonickém tvaru ve formátu jpeg a kódování base64.

1.2.5. Notifikace událostí

Přihlášení k odběru

Protokol	API
TCP	automaticky
HTTP	

XML zahájení události

```
<?xml version="1.0" encoding="utf-8"?>
<ateas>
  <event>
    <id>1</id>
    <imageid>1</imageid>
    <level>1</level>
    <server>
      <id>1</id>
      <name>Server 1</name>
    </server>
    <camera>
      <id>1</id>
      <name>Camera 1</name>
    </camera>
    <source>
      <id>1</id>
    </source>
    <datetime>
      <utcstamp>128989433710312500</utcstamp>
      <localvalue>1.7.2023 9:05:51</localvalue>
    </datetime>
    <data></data>
  </event>
</ateas>
```

```
<uuid></uuid>
<videoobject>
  <rectangle>20 20 200 200</rectangle>
</videoobject>
</event>
</ateas>
```

XML ukončení události

```
<?xml version="1.0" encoding="utf-8"?>
<ateas>
  <eventstop>
    <id>1</id>
    <datetime>
      <utcstamp>128989433710312500</utcstamp>
      <localvalue>1.7.2023 9:05:51</localvalue>
    </datetime>
    <data></data>
  </eventstop>
</ateas>
```

POZNÁMKA

Události jsou centralizovány ze všech kamerových serverů systému.

POZNÁMKA

Číslo obrázku spojuje danou událost s názvem souboru, který může být exportován na FTP server.

POZNÁMKA

Číslo zdroje jednoznačně určuje daný typ události, které lze zobrazit pomocí testovacího programu.

Příklady ID zdroje:

- 1 – detekce pohybu na kameře
- 2 – výpadek kamery
- 3 – poplachový vstup kamery, data obsahují číslo vstupu
- 10 – 14 - rozpoznání RZ vozidla, data obsahují RZ vozidla v dekorativním tvaru
- 32 – kontrola kvality videa, data obsahují požadovanou úroveň snímkové frekvence
- 40 – detekce pohybu na serveru
- 51 – 100 – vlastní události
- 110 – událost ručního nahrávání kamery
- 111 – 130 – Onvif události, data mohou obsahovat dodatečné informace
- 131 – 150 – komplexní události
- 151 – 200 – vlastní události
- 201 – 250 – analytické události

POZNÁMKA

UTC časové razítko udává absolutní čas výskytu události nezávislý na časové zóně a změnách letního času jako počet 100 nanovteřinových intervalů od 1. ledna 1601 UTC.

1.2.6. Notifikace přihlášení

Přihlášení k odběru

Protokol	API
TCP	automaticky
HTTP	

XML přihlášení uživatele

```
<?xml version="1.0" encoding="utf-8"?>
<ateas>
  <user>
    <id>10</id>
    <name>tester</name>
    <action>login</action>
```

```

<datetime>
  <utcstamp>128989433710312500</utcstamp>
  <localvalue>22.7.2023 15:05:23</localvalue>
</datetime>
</user>
</ateas>

```

POZNÁMKA

Při odhlášení má parametr akce hodnotu logout.

POZNÁMKA

UTC časové razítko udává absolutní čas výskytu události nezávislý na časové zóně a změnách letního času jako počet 100 nanovteřinových intervalů od 1. ledna 1601 UTC.

1.3. ATEAS API pro kamerový server

1.3.1. Externí události

Příjem externí události

Protokol	API
TCP	[ATEAS EVENT (status) (deviceid) (code) (data)]
HTTP	

Parametry:

Parametr	Hodnota	Výchozí	Význam
status	START, STOP		Zahájení či ukončení
deviceid	1 – 999		Číslo zařízení
code	(dle nastavení)		Název vlastní události

data	(max. 200 znaků)		Dodatečná data
-------------	------------------	--	----------------

Příklad:

Protokol	API
TCP	[ATEAS EVENT START 1 TEMPERATURE 76]
HTTP	

POZNÁMKA

Kód událostí musí odpovídat vytvořenému názvu v části administrace kamer.

POZNÁMKA

Událost může být ukončena explicitně anebo automaticky nastavením maximální doby trvání.

1.3.2. Metadata

Vkládání metadat

Protokol	API
TCP	[ATEAS META (deviceid) (timestamp) (code) (data)]
HTTP	

Parametry:

Parametr	Hodnota	Výchozí	Význam
deviceid	1 – 999		Číslo zařízení
timestamp	0 – N		UTC časové razítko
code	(dle nastavení)		Název vlastní události

data	(max. 200 znaků)		Metadata
-------------	------------------	--	----------

Příklad:

Protokol	API
TCP	[ATEAS META 1 0 SCAN AB512459]
HTTP	

POZNÁMKA

UTC časové razítko udává absolutní čas výskytu události nezávislý na časové zóně a změnách letního času jako počet 100 nanovteřinových intervalů od 1. ledna 1601 UTC.

POZNÁMKA

Časové razítko může mít hodnotu 0. V takovém případě bude čas určen serverem. Vlastní zadání času je vhodné při offline nahrávání dat do systému, při kterém není akceptován čas menší než 30 dní v minulosti a větší než 1 minutu v budoucnosti dle času serveru.

POZNÁMKA

Kód událostí musí odpovídat vytvořenému názvu v části administrace kamer.

1.4. Parametrické spouštění aplikací

Aplikace ATEAS Security je možné spouštět také s dodatečnými parametry, které jsou předány spustitelnému souboru při startu aplikace či služby. Přidání těchto parametrů je možné v nastavení služby v systému Windows. V současné době existují parametry popsané níže.

1.4.1. Administrační server

Parametr	Hodnoty	Význam	Poznámka
----------	---------	--------	----------

-ssl	heslo	Heslo certifikátu	Je nutné zadat, pokud je použitý certifikát ve formátu PFX chráněn heslem.
-------------	-------	-------------------	--

1.4.2. Kamerový server

Parametr	Hodnoty	Význam	Poznámka
-ssl	heslo	Heslo certifikátu	Je nutné zadat, pokud je použitý certifikát ve formátu PFX chráněn heslem.
-loglevel	0 - 1	Nastavení úrovně logování	Pozitivní hodnota zapíná výpis zaplnění vyrovnávací paměti serveru do podsložky log.