



Cisco ASA REST API 빠른 시작 설명서

최초 발행: 2014년 12월 24일

수정일: 2015년 6월 2일

목차

개요, 1페이지

ASA REST API 요청 및 응답, 2페이지

ASA REST API 에이전트 및 클라이언트의 설치와 구성, 3페이지

문서 콘솔 및 API 스크립트 내보내기, 5페이지

ASA에서 REST API 디버깅 활성화, 6페이지

ASA REST API 관련 Syslog 메시지, 7페이지

관련 설명서, 8페이지

개요

개별 Cisco ASA를 구성 및 관리하는 데 사용할 수 있는 몇 가지 옵션이 있습니다.

- CLI(Command Line Interface) - 연결된 콘솔을 통해 ASA에 제어 명령을 직접 전송합니다.
- ASDM(Adaptive Security Device Manager) - ASA의 구성, 관리 및 모니터링에 사용할 수 있는 GUI가 포함된 "내장형" 관리 애플리케이션입니다.
- Cisco Security Manager - 여러 보안 디바이스로 구성된 중간 및 대규모 네트워크를 대상으로 하는 이 그래픽 애플리케이션을 통해 개별 ASA를 구성, 관리 및 모니터링할 수 있습니다.

Cisco ASA REST API가 출시됨에 따라 이제 사용하기 쉬운 경량의 옵션이 하나 더 추가되었습니다. 이것은 API가 실행 중인 ASA에서 빠르게 다운로드 및 활성화할 수 있는, "RESTful" 원칙을 기반으로 하는 API(application programming interface)입니다.

브라우저에 REST 클라이언트를 설치한 후에는 특정 ASA의 REST 에이전트에 연결하고, 표준 HTTP 메서드를 사용하여 현재 컨피그레이션 정보에 액세스하고, 추가 컨피그레이션 매개 변수를 사용할 수 있습니다.





주의

ASA에서 REST API가 활성화된 경우 다른 보안 관리 프로토콜에 의한 연결이 차단되지 않습니다. 즉, 자신이 동일한 작업을 수행하는 동안 CLI, ASDM 또는 Security Manager를 사용하는 다른 사용자가 ASA 컨피그레이션을 변경할 수 있습니다.

ASA REST API 요청 및 응답

ASA REST API는 REST(Representational State Transfer) API를 통해 개별 ASA를 관리하기 위한 프로그램 방식의 액세스를 제공합니다. API를 사용하면 외부 클라이언트가 ASA 리소스에 대해 CRUD(Create, Read, Update, Delete) 작업을 수행할 수 있습니다. HTTPS 프로토콜 및 REST 방법론이 기반이 됩니다.

모든 API 요청은 HTTPS를 통해 ASA에 전송되고 응답이 반환됩니다.

이 섹션에서는 요청이 구성되는 방법 및 예상 응답에 대한 개요를 제공합니다.

요청 구조

사용 가능한 요청 메서드:

- GET - 지정된 개체에서 데이터를 검색합니다.
- PUT - 제공된 정보를 지정된 개체에 추가합니다. 개체가 없을 경우 404 Resource Not Found(404 리소스를 찾을 수 없음) 오류가 반환됩니다.
- POST - 제공된 정보로 개체를 만듭니다.
- DELETE - 지정된 개체를 삭제합니다.
- PATCH - 지정된 개체에 부분 수정을 적용합니다.

응답 구조

각 요청은 표준 헤더, 응답 콘텐츠 및 상태 코드로 ASA에서 HTTPS 응답을 생성합니다.

응답 구조는 다음과 같을 수 있습니다.

- LOCATION - 새로 만든 리소스 ID. POST 전용 - 새 리소스 ID 보유(URI 표현으로서).
- CONTENT-TYPE - 응답 메시지 본문을 설명하는 미디어 유형. 응답 메시지 본문의 표현 및 구문을 설명.

각 응답은 HTTP 상태 또는 오류 코드를 포함합니다. 사용 가능한 코드의 범주는 다음과 같습니다.

- 20x - 다음을 비롯한 200 계열 코드는 성공적인 작업을 나타냅니다.
 - 200 OK - 성공적인 요청의 표준 응답.
 - 201 Created - 요청이 완료되었으며 새 리소스가 생성됨.
 - 202 Accepted - 요청이 수락되었지만 처리가 완료되지 않음.
 - 204 No Content - 서버가 성공적으로 요청을 처리했으며 콘텐츠가 반환되지 않음.
- 4xx - 다음을 비롯한 400 계열 코드는 클라이언트 측 오류를 나타냅니다.
 - 400 Bad Request - 인식할 수 없는 매개 변수, 누락된 매개 변수 또는 잘못된 값을 비롯한 유효하지 않은 쿼리 매개 변수.

- 404 Not Found - 제공된 URL이 기존 리소스와 일치하지 않음. 예를 들어, 리소스를 사용할 수 없으므로 HTTP DELETE가 실패할 수 있습니다.
- 405 Method not Allowed - 리소스에서 허용되지 않는 HTTP 요청이 표시되었습니다(예: 읽기 전용 리소스의 POST).

• 5xx - 500 계열 코드는 서버 측 오류를 나타냅니다.

오류가 발생하면 오류 코드 외에도 오류에 대해 자세히 설명하는 오류 개체가 반환 응답에 포함될 수 있습니다. JSON 오류/경고 응답 스키마는 다음과 같습니다.

```
[
  {
    "code"      : "string",
    "details"   : "string",
    "context"  : attribute name,
    "level"    : <Error/Warning/Info>
  },
  ...
]
```

개체 속성은 다음과 같습니다.

속성	유형	설명
messages	사전 목록	오류 또는 경고 메시지 목록
code	문자열	오류/경고/정보 코드
details	문자열	오류/경고/정보에 해당하는 자세한 메시지



참고

REST API 호출에 의한 ASA 컨피그레이션의 변경 사항은 시동 컨피그레이션에 남아 있지 않게 됩니다. 즉, 변경 사항은 실행 중인 컨피그레이션에만 할당됩니다. 따라서 변경 사항을 시동 컨피그레이션에 저장하려면 writemem API 요청을 게시(POST)할 수 있습니다. 자세한 내용은 [Write Memory API](#)를 참조하십시오.

ASA REST API 에이전트 및 클라이언트의 설치와 구성

다음 섹션에서는 ASA에서 REST API 에이전트를 설치 및 활성화하고, 로컬 호스트에서 REST API 클라이언트를 구성하는 방법에 대해 설명합니다.

REST API 에이전트 다운로드 및 설치

CLI를 사용하여 다음 단계를 수행하여 특정 ASA에서 ASA REST API 에이전트를 다운로드 및 설치합니다.

단계 1 원하는 ASA에서 `copy <package> disk0:` 명령을 실행하여 cisco.com에서 ASA의 플래시 메모리로 현재 ASA REST API 패키지를 다운로드합니다. 예를 들면 다음과 같습니다.

```
copy tftp://10.7.0.80/asa-restapi-111-1fbff-k8.SPA disk0:
```

단계 2 `rest-api image disk0:/<package>` 명령을 실행하여 패키지를 확인하고 설치합니다. 예:

```
rest-api image disk0:/asa-restapi-111-1fbff-k8.SPA
```

설치 프로그램은 호환성 및 검증 확인을 수행한 다음 패키지를 설치합니다. ASA는 재부팅되지 않습니다.

REST API 에이전트 활성화 및 구성

다음 단계를 수행하여 특정 ASA에서 ASA REST API 에이전트를 활성화 및 구성합니다.

- 단계 1** 정확한 소프트웨어 이미지가 ASA에 설치되었는지 확인합니다.
- ASA 55nn-X의 경우 asa100-12-16-15-smp-k8.bin 이상입니다. ASA의 경우 asav100-12-16-15.ova입니다. .ova 이미지가 ASA에 이미 설치된 경우 asa100-12-16-15-smp-k8.bin을 사용하여 VM 이미지를 업데이트할 수 있습니다.
- 단계 2** CLI를 사용하여, HTTP 서버가 ASA에서 활성화되었는지와 API 클라이언트가 관리 인터페이스에 연결할 수 있는지를 확인합니다. 예를 들면 다음과 같습니다.
- ```
http server enable
http 0.0.0.0 0.0.0.0 <management interface nameif>
```
- 단계 3** CLI를 사용하여 API 연결을 위한 HTTP 인증을 정의합니다. 예를 들면 다음과 같습니다.
- ```
aaa authentication http console LOCAL
```
- 단계 4** CLI를 사용하여 읽기/쓰기 권한으로 로컬 사용자를 만듭니다(이 경우 권한 레벨은 15). 예를 들면 다음과 같습니다.
- ```
username <user name> password <password> encrypted privilege 15
```
- 권한 레벨 3 이상은 모니터링 요청을 호출하는 데 필요하고, 권한 레벨 5 이상은 GET 요청을 호출하는 데 필요하며, 권한 레벨 15는 PUT/POST/DELETE 작업을 호출하는 데 필요합니다.
- 단계 5** CLI를 사용하여 ASA에 API 트래픽을 위한 고정 경로를 만듭니다. 예를 들면 다음과 같습니다.
- ```
route <management interface nameif> 0.0.0.0 0.0.0.0 <gwip> 1
```
- 단계 6** CLI를 사용하여 ASA에서 ASA REST API 에이전트를 활성화합니다. 예를 들면 다음과 같습니다.
- ```
rest-api agent
```

## REST API 클라이언트 구성

다음 단계를 수행하여 로컬 호스트 브라우저에서 REST API 클라이언트를 설치 및 구성합니다.

- 단계 1** 브라우저용 REST API 클라이언트를 다운로드 및 설치합니다.
- Chrome의 경우 Google에서 REST 클라이언트를 설치합니다. Firefox의 경우 RESTClient 애드온을 설치합니다. Internet Explorer는 지원되지 않습니다.
- 단계 2** 브라우저를 사용하여 다음 요청을 시작합니다.
- ```
https:<asa management ip address>/api/objects/networkobjects
```
- 오류 없음 응답을 수신하는 경우 ASA에서 작동하는 REST API 에이전트에 도달한 것입니다.

에이전트 요청에 문제가 있는 경우 [ASA에서 REST API 디버깅 활성화](#), 6페이지에 설명된 대로 CLI 콘솔에 디버깅 정보 표시 기능을 활성화할 수 있습니다.

단계 3 선택적으로, POST 작업을 수행하여 ASA에 대한 연결을 테스트할 수 있습니다.

예를 들면 다음과 같습니다.

기본 인증 자격 증명 (<username><password>) 또는 인증 토큰(자세한 내용은 [Token Authentication 참조](#))을 제공합니다.

대상 요청 주소: `https://<asa management ipaddress>/api/objects/networkobjects`

본문 콘텐츠 유형: `application/json`

가공되지 않은 작업 본문:

```
{
  "kind": "object#NetworkObj",
  "name": "TestNetworkRangeObj",
  "host": {
    "kind": "IPv4Network",
    "value": "12.12.12.0/24"
  }
}
```

이제 ASA REST API를 사용하여 ASA를 구성 및 모니터링할 수 있습니다. 통화 설명 및 예는 API 설명서를 참조하십시오.

문서 콘솔 및 API 스크립트 내보내기

ASA에서 API 호출에 대해 알아보거나 API 호출을 직접 시도하기 위한 "샌드박스"로서, `host:port/doc`에 있는 REST API 온라인 문서 콘솔("Doc UI")을 사용할 수 있습니다.

또한 Doc UI의 **Export Operation(작업 내보내기)** 버튼을 사용하여, 표시된 메서드 예를 로컬 호스트에 JavaScript, Python 또는 Perl 스크립트 파일로 저장할 수 있습니다. 그런 다음 사용 중인 ASA에 이 스크립트를 적용할 수도 있고, 다른 ASA 및 기타 네트워크 디바이스의 애플리케이션에 맞게 수정할 수도 있습니다. 기본 용도는 교육 및 부트스트래핑 톨로 사용하는 것입니다.

JavaScript

JavaScript 파일을 사용하려면 `node.js`를 설치해야 합니다(<http://nodejs.org/>). `node.js`를 사용하면 JavaScript 파일(대개 브라우저용으로 작성됨)을 명령행 스크립트처럼 사용할 수 있습니다. 설치 지침을 따른 후 `node script.js`로 스크립트를 실행하면 됩니다.

Python

Python 스크립트를 사용하려면 Python을 설치해야 합니다(<https://www.python.org/>). Python을 설치했으면 `python script.py username password`로 스크립트를 실행할 수 있습니다.

Perl

Perl 스크립트를 사용하려면 몇 가지 추가 설정이 필요합니다. Perl 자체 및 4개의 Perl 라이브러리, 이렇게 다섯 개의 구성 요소가 필요합니다.

- Perl 패키지 - <http://www.perl.org/>
- Bundle::CPAN - <http://search.cpan.org/~andk/Bundle-CPAN-1.861/CPAN.pm>
- REST::Client - <http://search.cpan.org/~mcrawfor/REST-Client-88/lib/REST/Client.pm>
- MIME::Base64 - <http://perldoc.perl.org/MIME/Base64.html>
- JSON - <http://search.cpan.org/~makamaka/JSON-2.90/lib/JSON.pm>

다음은 Macintosh의 Perl 부트스트래핑 예입니다.

```
$ sudo perl -MCPAN e shell
cpan> install Bundle::CPAN
cpan> install REST::Client
cpan> install MIME::Base64
cpan> install JSON
```

종속 요소를 설치한 후 `perl script.pl username password`를 사용하여 스크립트를 실행할 수 있습니다.

ASA에서 REST API 디버깅 활성화

ASA에서 REST API의 구성 또는 연결에 문제가 있는 경우 다음 CLI 명령을 사용하여 콘솔에 디버깅 메시지 표시 기능을 활성화할 수 있습니다. 디버그 메시지를 비활성화하려면 명령의 **no** 형식을 사용합니다.

```
debug rest-api [agent | cli | client | daemon | process | token-auth] [error | event]
no debug rest-api
```

구문 설명

agent	(선택 사항) REST API 에이전트 디버깅 정보를 활성화합니다.
cli	(선택 사항) REST API CLI 디먼-에이전트 통신을 위한 디버깅 메시지를 활성화합니다.
client	(선택 사항) REST API 클라이언트 및 REST API 에이전트 간 메시지 라우팅을 위한 디버깅 정보를 활성화합니다.
daemon	(선택 사항) REST API CLI 디먼-에이전트 통신을 위한 디버깅 메시지를 활성화합니다.
프로세스	(선택 사항) REST API 에이전트 프로세스 시작/중지 디버깅 정보를 활성화합니다.
token-auth	(선택 사항) REST API 토큰 인증 디버깅 정보.
오류	(선택 사항) API에서 로깅한 오류로 디버그 메시지를 제한하려면 이 키워드를 사용합니다.
event	(선택 사항) API에서 로깅한 이벤트로 디버그 메시지를 제한하려면 이 키워드를 사용합니다.

사용 지침

특정 구성 요소 키워드를 제공하지 않으면(즉, 단순히 **debug rest-api** 명령만 실행하면) 디버그 메시지가 모든 구성 요소 유형에 대해 표시됩니다. **event** 또는 **error** 키워드를 제공하지 않으면 지정된 구성 요소에 대해 이벤트 및 오류 메시지가 모두 표시됩니다. 예를 들어 **debug rest-api daemon event**는 API 디먼-에이전트 통신에 대한 이벤트 디버그 메시지만 표시합니다.

관련 명령

명령	설명
debug http	HTTP 트래픽에 대한 자세한 정보를 표시하려면 이 명령을 사용합니다.

ASA REST API 관련 Syslog 메시지

이 절에서는 ASA REST API 관련 시스템 로그 메시지에 대해 설명합니다.

342001

오류 메시지 %ASA-7-342001: REST API Agent started successfully.

설명 REST API 클라이언트가 ASA를 구성할 수 있으려면 먼저 REST API 에이전트를 성공적으로 시작해야 합니다.

권장 조치 없습니다

342002

오류 메시지 %ASA-3-342002: REST API Agent failed, reason: *reason*

설명 REST API 에이전트가 여러 이유로 시작하지 못하거나 충돌하며, 그 이유가 지정됩니다.

- *reason* - REST API 실패의 원인

권장 조치 문제를 해결하기 위해 수행하는 작업은 로깅된 이유에 따라 달라집니다. 예를 들어, Java 프로세스에 메모리가 부족하면 REST API 에이전트가 충돌합니다. 이런 경우 REST API 에이전트를 재시작해야 합니다. 재시작에 성공하지 못하면 Cisco TAC에 문의하여 근본 원인 해결 방법을 확인하십시오.

342003

오류 메시지 %ASA-3-342003: REST API Agent failure notification received. Agent will be restarted automatically.

설명 REST API 에이전트에서 실패 알림이 수신되고 에이전트 재시작이 시도됩니다.

권장 조치 없습니다

342004

오류 메시지 %ASA-3-342004: Failed to automatically restart the REST API Agent after 5 unsuccessful attempts. Use the 'no rest-api agent' and 'rest-api agent' commands to manually restart the Agent.

설명 REST API 에이전트가 여러 번의 시도 후 시작에 실패했습니다.

권장 조치 실패한 이유를 더 잘 알아보려면 syslog %ASA-3-342002를 참조하십시오(로깅된 경우). REST API 에이전트를 비활성화하려면 **no rest-api agent** 명령을 사용하고, REST API 에이전트를 다시 활성화하려면 **rest-api agent** 명령을 사용해보십시오.

관련 설명서

ASA 및 ASA의 컨피그레이션과 관리에 대해 자세히 알아보려면 다음 링크를 사용하십시오.

- *Cisco ASA Series 설명서 찾아보기*: <http://www.cisco.com/go/asadocs>

ASAv에서 지원되지 않는 ASA 기능 목록을 보려면 다음 링크를 사용하십시오.

- <http://www.cisco.com/c/en/us/td/docs/security/asa/asa92/configuration/general/asa-general-cli/intro-asav.html#pgfId-1156883>

이 문서는 “관련 설명서” 섹션에 나와 있는 문서와 함께 사용해야 합니다.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

이 문서에서 사용된 모든 IP(인터넷 프로토콜) 주소와 전화 번호는 실제 주소와 전화 번호가 아닙니다. 이 문서에 포함된 예제, 명령 표시 출력, 네트워크 토폴로지 다이어그램 및 다른 그림은 이해를 돕기 위한 자료일 뿐이며, 실제 IP 주소나 전화 번호가 사용되었다면 이는 의도하지 않은 우연의 일치입니다.

© 2014-2015 Cisco Systems, Inc. 모든 권한은 당사에 있습니다.