01 용어 정리

용어 정리

Security Management은 정책을 만들고 게이트웨이에 내려보내는 "지휘 본부" 입니다. 이 가이드를 읽을 때 바탕이 되는 핵심 용어만 골라 흐름에 따라 풀어 둡니다.

관리의 세 축과 데이터

가장 먼저 **Security Management Server** 입니다. 객체·정책을 관리·저장하고 게이트웨이에 배포하며 보안 이벤트를 모니터링 하는 서버입니다. 이를 다루는 GUI가 **SmartConsole**, 정책을 실제로 집행하는 것이 **Security Gateway** 입니다(소개).

관리 서버가 다루는 것이 **객체(Object)** 와 **정책(Policy)** 입니다. **Network Object** 는 컴퓨터·네트워크·주소 범위·서비스 등 토폴로지의 여러 부분을 나타내는 논리 객체 이고, **Rule** 은 트래픽 조건과 동작 한 줄, 그 묶음 전체가 **Rule Base** 입니다. 여러 정책 종류를 한데 묶어 함께 설치하는 단위가 **Policy Package** 입니다(정책 관리 기초).

R80부터 정책은 **Session 단위** 로 다룹니다. 변경을 **Publish(게시)** 해야 다른 관리자에게 보이고 설치 준비가 되며, 각 publish가 **Revision(리비전)** 을 만들어 과거로 되돌릴 수 있습니다.

사용자와 관리자, 인증

Administrator(관리자) 는 SmartConsole·CLI·API로 보안 환경을 관리하는 사람 이고, **User(사용자)** 는 환경에서 트래픽을 일으키는 객체 입니다. 관리자는 Management Server가 인증하고, 사용자는 Security Gateway가 인증 한다는 구분이 중요합니다(관리자·사용자 계정 관리).

장비 사이 통신의 토대는 **SIC(Secure Internal Communication)** 입니다. Check Point 장비들이 SSL로 서로를 인증하는 메커니즘 으로, **ICA(Internal Certificate Authority)** 가 발급한 인증서에 기반합니다. ICA는 Management Server에 내장된 인증 기관 입니다(인증 인프라).

사용자 디렉터리 연동에는 **User Directory** 블레이드(LDAP 연동)와 **Active Directory** 가 쓰이고, 권한은 **Permission Profile(권한 프로파일)** 로 관리자에게 배정합니다.

운영·확장 용어

Management High Availability 는 관리 서버의 이중화·데이터베이스 백업 입니다. 첫 설치 서버가 **Primary**, 이후가 **Secondary** 이며, 한 대가 **Active**·나머지가 **Standby** 로 동기화됩니다(관리 서버 운영).

규모가 커지면 **Multi-Domain Security Management** 로 여러 도메인(Domain Management Server)을 한 서버에서 관리합니다. 로그는 전용 **Log Server**, 이벤트 분석은 **SmartEvent** 가 맡습니다. 자동화에는 **Management API**(`mgmt_cli` 등)를 씁니다(API로 관리).

이 밖에 정책에서 자주 쓰는 객체로 **Updatable Object**(Microsoft 365·AWS·Geo 위치 등 외부 서비스를 나타냄), **Network Feed**(외부 서버가 제공하는 IP·도메인 목록), **Access Role**(신원 기반 규칙에 쓰는 객체)이 있습니다.

02 Security Management 소개

Security Management 소개

Security Gateway 가이드가 정책을 집행하는 엔진 을 다뤘다면, 이 가이드는 그 정책을 만들어 게이트웨이에 내려보내는 "지휘 본부" — Security Management Server를 다룹니다. 이 장에서는 전체 구성과 작업의 큰 줄기를 잡습니다.

이 가이드는 기본 Security Management Server 배포 에 초점을 둡니다. 여러 사이트를 거느린 조직이라면 R82 Multi-Domain Security Management 관리자 가이드를 참고하세요.

!Check Point 보안 아키텍처 *① SmartConsole — Management Server에 접속·관리하는 GUI ② Security Management Server — 정책으로 게이트웨이를 관리하고 이벤트를 모니터링 ③ Security Gateway — 네트워크 경계에 두어 정책을 집행 ④ 보호할 내 환경*

세 가지 기본 구성요소

큰 그림은 단순합니다. **SmartConsole**(관리 GUI)으로 정책을 만들면, **Security Management Server**(관리 서버)가 그것을 저장·관리하고, **Security Gateway**(게이트웨이)가 경계에서 집행 합니다. 관리 서버는 정책을 내려보낼 뿐 아니라 네트워크의 보안 이벤트도 모니터링합니다.

보안 관리 구성의 작업 흐름

처음 환경을 세우는 큰 줄기는 여덟 걸음 으로 흐릅니다.

먼저 SmartConsole로 관리 서버에 로그인 하고(SmartConsole 이해), 관리 서버와 게이트웨이를 구성 합니다(계획·구성). 그다음 환경의 관리자를 정의하고 권한을 배정 하며, 보호 대상이 되는 사용자·사용자 그룹을 정의 합니다(관리자·사용자 계정 관리).

이어서 물리·가상 네트워크 구성요소를 객체로 구성 하고, 조직 자원을 보호하는 접근 규칙을 정의 한 뒤(Access Control 정책 만들기), 마지막으로 정책을 설치(Install Policy) 해 게이트웨이가 집행하게 합니다.

정리하면 로그인 → 서버·게이트웨이 구성 → 관리자·사용자 정의 → 객체 구성 → 접근 규칙 정의 → 정책 설치 의 한 줄기이며, 이 가이드의 각 장은 이 흐름의 한 단계씩을 자세히 풀어 줍니다.

03 SmartConsole 이해

SmartConsole 이해

SmartConsole 은 복잡한 네트워크의 보안을 다루는 Check Point의 관리 GUI 입니다. 정책을 짜기 전에 이 도구의 짜임새부터 익혀 두면 뒤 장들이 한결 수월합니다. 이 장은 SmartConsole 창의 구성, 검색·도구, 그리고 접속 방법을 정리합니다.

SmartConsole 창

!SmartConsole 창 구성 *① Global Toolbar ② Session Management Toolbar ③ Navigation Toolbar ④ Objects Bar(F11) ⑤ Validations 창 ⑥ 명령줄 인터페이스 버튼*

화면은 크게 위쪽 툴바, 왼쪽 내비게이션, 오른쪽 객체·검증 창 으로 나뉩니다. **Global Toolbar**(맨 위)에는 정책·레이어 관리, 객체 생성, 세션 게시/취소, 정책 설치, Global Properties 등 주요 메뉴가 모여 있습니다. 그 아래 **Session Management Toolbar** 에서 변경을 게시(Publish)하거나 취소(Discard) 합니다 — 게시해야 다른 관리자에게 변경이 보이고, 정책 설치 시 그 게시된 변경이 게이트웨이에 적용 됩니다.

왼쪽 **Navigation Toolbar** 가 작업 공간을 가릅니다. **Gateways & Servers**(Ctrl+1)는 게이트웨이 관리·블레이드 활성화·상태 확인, **Security Policies**(Ctrl+2)는 Access Control·Threat Prevention·공유 정책 편집, **Logs & Events**(Ctrl+3)는 로그 검색·리포트·모니터링·규정 준수, **Infinity Services**(Ctrl+4)는 Infinity Portal 연동, **Manage & Settings**(Ctrl+5)는 관리자·권한·신뢰 클라이언트·환경설정 입니다. 맨 아래 **명령줄 버튼**(F9)으로 관리 스크립팅·API 창을 엽니다.

검색 엔진

각 뷰에서 관리 서버 데이터베이스를 검색 할 수 있습니다. 객체는 이름 앞부분을 입력(예: `USG` 로 USGlobalHost)하거나, `*` 를 붙여 부분 문자열(예: `*host`) 로 찾습니다.

IP 검색은 두 모드입니다. **General**(기본)은 직접·간접 매칭(IP를 가진 객체, 그 객체를 포함한 그룹·네트워크·범위, 그것을 쓰는 규칙)을 모두 찾고, **Packet** 은 그 IP를 가진 패킷이 게이트웨이에 도착한 것처럼 매칭되는 규칙 을 보여 줍니다(`mode:Packet` , 단 IPv6 미지원).

도구와 공유 정책

Access Control 뷰의 **Access Tools** 와 Threat Prevention 뷰의 **Custom Policy Tools** 가 추가 관리·데이터 수집 도구를 제공합니다. VPN Communities·Updates·Client Certificates·Application Wiki·Installation History 같은 도구, 그리고 Threat 쪽의 Profiles·IPS Protections·Indicators·Threat Wiki 등입니다.

Shared Policies 는 Policy Package에 속하지 않고 모든 패키지가 공유하는 정책 으로, Mobile Access·DLP·HTTPS Inspection·Inspection Settings가 여기 속하며 Access Control 정책과 함께 설치됩니다.

객체와 규칙은 API 명령줄(F9)로도 다룰 수 있 어, 스크립트로 구성·운영을 자동화할 수 있습니다(API로 관리).

접속 방법

SmartConsole에 로그인하려면 관리 서버에 관리자 계정 이 있어야 합니다(첫 관리자는 설치 시 First Time Configuration Wizard에서 생성). 로그인은 사용자명/암호, 인증서 파일, CAPI 인증서, Identity Provider(SAML) 중에서 선택하며, 서버 이름·IP를 입력해 접속합니다. 읽기 전용(Read Only)으로 접속 할 수도 있고, 처음 접속 때는 fingerprint를 확인 해 신뢰를 굳힙니다.

브라우저로 쓰는 **Web SmartConsole** 도 있습니다. `https://<관리 서버 IP>/smartconsole` 로 접속하며(Chrome 권장), SmartConsole GUI 기능을 웹에서 제공합니다.

04 보안 관리 계획·구성

보안 관리 계획·구성

환경을 세우기 전에 관리 서버의 크기를 가늠하고, 토폴로지·접근 정책을 설계 한 뒤, 관리 서버와 게이트웨이를 구성 하는 것이 순서입니다. 이 장은 그 계획과 첫 구성을 정리합니다.

관리 서버 사이징

관리 서버를 제대로 키우려면 관리할 게이트웨이 수, 초당 지속 로그 수 가 기본 데이터이고, 대규모라면 도메인 수·Rule Base 크기·동시 관리자 수 까지 봅니다(가이드라인은 sk181782, 대규모는 sk178325).

권장 사항은 분리입니다. 전용 Management Server·전용 Log Server를 쓰고, Management High Availability를 구성 하며(standby 서버를 게이트웨이의 Log Server로 활용 가능), 전용 SmartEvent Server 를 둡니다.

보안 환경 설계의 세 걸음

설치를 마쳤다면 사이버 보안 구성으로 이어집니다.

먼저 조직의 토폴로지를 정의 합니다. 물리·가상 게이트웨이, 호스트, 서버, 서비스, 네트워크, 주소 범위, 그룹 등 각 구성요소를 SmartConsole의 객체로 만드는 것입니다. 함께 보호 대상인 사용자·사용자 그룹 도 정의하는데, 직접 입력하거나 LDAP·User Directory·Active Directory 연동 으로 추가합니다(관리자·사용자 계정 관리).

다음으로 자원을 보호할 접근 규칙을 정의하고 정책으로 묶 습니다. 트래픽·애플리케이션·웹사이트·데이터 기준으로 정책을 짜고(정책 관리 기초), Anti-Virus·Anti-Malware로 알려진 위협에 선제 대응하며, UserCheck로 사용자를 교육하고, 로깅·모니터링으로 트래픽과 이벤트를 추적합니다.

마지막으로 접근 정책을 집행 합니다 — 게이트웨이를 구성하고 적절한 Software Blade를 켠 뒤 정책을 설치 하는 것입니다.

관리 서버와 게이트웨이 구성

관리 서버 구성 은 SmartConsole의 **Gateways & Servers** 에서 관리 서버 객체를 더블클릭해, **Management 탭** 에서 필요한 Software Blade를 켜는 것입니다. **Network Policy Management**(통합 보안 정책 관리, 자동 활성) 가 기본이고, 여기에 **Endpoint Policy Management**(켜면 끌 수 없음), **Logging & Status**, **Identity Logging**, **User Directory**, **Provisioning**, **Compliance**, **SmartEvent Server / Correlation Unit** 등을 필요에 따라 켭니다.

게이트웨이 구성 은 **Gateways & Servers** > **New** > **Gateway** 에서 시작하며, **Wizard Mode**(마법사)와 **Classic Mode**(상세 구성) 중에 고릅니다. 게이트웨이를 실제로 만들고 관리하는 자세한 절차는 게이트웨이 만들기·관리에서 다룹니다.

05 API로 관리

API로 관리

SmartConsole로 손수 하는 일은 API로 자동화 할 수 있습니다. 이 장은 관리 서버의 API Server, 그것을 다루는 도구, 그리고 API 키 인증을 정리합니다.

API Server와 도구

관리 서버에는 **API Server** 가 돌아, API 요청을 보내 서버를 구성·제어 할 수 있습니다. 일상 작업을 자동화하고, 가상화 서버·티켓팅·변경 관리 같은 서드파티 시스템과 Check Point를 연동 하는 스크립트를 돌립니다. API 문서는 온라인(Check Point Management API Reference)이나 로컬(`https://<서버 IP>/api_docs` , 기본 비활성)에서 봅니다.

API를 다루는 도구는 세 가지입니다. Gaia에 포함된 `mgmt_cli` , SmartConsole에 포함된 `mgmt_cli.exe` (다른 Windows PC로 복사 가능), 그리고 HTTP로 통신하는 Web Services API(`https://<서버 IP>/web_api/<command>`) 입니다.

API Server 구성

설정은 **Manage & Settings** > **Blades** > **Management API** > **Advanced Settings** 에서 합니다. **Startup**(서버 시작·재부팅 시 자동 시작 — RAM 4GB 초과면 기본 활성) 과 **Access**(어떤 클라이언트가 접속할지) 를 정합니다.

Access는 세 단계입니다. **Management server only**(서버 자신만, `mgmt_cli` 만 가능), **All IP addresses that can be used for GUI clients**(Trusted Clients에 정의된 IP — SmartConsole·Web· `mgmt_cli`), **All IP addresses**(모든 IP) 중에 고릅니다. 그다음 각 관리자의 Permission Profile에 **Management API Login** 권한 이 있는지 확인하고, 세션을 게시한 뒤 `api restart` 로 API Server를 재시작(Multi-Domain은 `mdsenv` 로 컨텍스트 지정), `api status` 로 상태를 확인 합니다.

> 참 고
>
> Access를 "GUI clients용 모든 IP"로 두고 Trusted Client가 200개를 넘으면 API 프로세스가 "Stopped"로 보일 수 있습니다.

API 키 인증

API key 는 API 호출 때 제시하는 토큰 으로, 사용자명/암호 대신 인증에 씁니다(이 관리자는 API 실행 전용, SmartConsole 인증에는 못 씀). **Manage & Settings** > **Permissions & Administrators** > **Administrators** 에서 새 관리자를 만들고, Authentication Method를 **API Key** 로 골라 **Generate API key** 한 뒤 키를 복사해 둡니다.

쓰는 방법은 간단합니다. `mgmt_cli login api-key <키>` 로 로그인해 토큰을 파일로 저장 한 뒤, `mgmt_cli -s <토큰파일> add simple-gateway …` 처럼 `-s` 플래그로 명령 을 실행합니다. API 키에 더해 인증서 파일을 함께 구성 하면, 관리자는 API 키나 인증서 중 하나로 인증할 수 있습니다.

```
mgmt_cli login api-key <api-key> > /var/tmp/token.txt
mgmt_cli -s /var/tmp/token.txt add simple-gateway name "gw1" ip-address 192.1
```

06 관리자·사용자 계정 관리

관리자·사용자 계정 관리

이 장은 누가 환경을 관리하고(관리자), 누구를 보호하는지(사용자) 를 정의·인증·관리하는 방법을 다룹니다. 원문은 분량이 가장 큰 장 중 하나라, 여기서는 핵심 개념과 인증 방식의 갈래 를 잡습니다. 세부 절차는 SmartConsole의 각 마법사를 따라가면 됩니다.

관리자와 사용자, 무엇이 다른가

관리자(Administrator) 는 SmartConsole·CLI·API로 보안 환경을 관리하는 IT 담당자 이고, **사용자(User)** 는 환경에서 트래픽을 일으키는 객체 입니다. 가장 중요한 구분은 누가 인증하느냐 입니다 — **관리자는 Security Management Server가, 사용자는 Security Gateway가** 인증합니다.

사용자는 모두 SmartConsole에서 직접 정의되어 관리 데이터베이스에 저장 되며(외부 AD 등에 정의된 사용자와 대비), 관리자가 정책을 설치할 때 관련 사용자 데이터가 게이트웨이로 복사 됩니다. 사용자는 접근 규칙(Remote Access VPN·Identity Awareness 등)에서 쓰여, 인가된 사용자에게만 자원 접근을 허용 함으로써 네트워크를 지킵니다.

사용자 계정과 인증 방식

사용자 계정을 만들 때는 인증 방식 을 고릅니다. **Check Point Password**(SmartConsole에 설정하는 정적 암호, 게이트웨이 로컬 DB에 저장) 가 가장 단순하고, 이 밖에 OS Password, RADIUS, TACACS, 인증서 기반 방식이 있습니다. 외부 디렉터리를 쓰면 LDAP·User Directory 연동이나 Microsoft Active Directory 로 사용자를 가져옵니다.

사용자는 사용자 그룹 으로 묶어 규칙의 Source로 쓸 수 있고(사용자는 자신이 속한 그룹을 알지 못함), 외부 그룹의 변경은 정책 설치 또는 사용자 데이터베이스 다운로드 후에야 적용됩니다.

관리자 계정과 권한

관리자도 여러 인증 방식 으로 만듭니다 — 사용자명/암호, 인증서 파일, CAPI 인증서, 그리고 SAML 기반 Identity Provider 로그인 등입니다. 관리자의 권한은 **Permission Profile(권한 프로파일)** 로 정해 배정합니다.

권한 프로파일은 관리자가 무엇을 볼·바꿀 수 있는지 를 세밀하게 정합니다. 예를 들어 API로 관리에서 본 Management API Login, Gaia API Proxy에서 쓰는 Run One Time Script, 모바일 인증서 발급만 허용하고 나머지는 제한하는 식의 역할 분리 가 가능합니다. 이렇게 업무별로 권한을 쪼개 최소 권한 원칙 을 지키는 것이 핵심입니다.

정리하면, 사용자는 보호 대상으로서 게이트웨이가 인증하고, 관리자는 운영 주체로서 관리 서버가 인증 하며, 각자에게 적절한 인증 방식과 권한 을 부여하는 것이 이 장의 요지입니다. 인증서의 바탕이 되는 ICA(내부 인증 기관)는 인증 인프라에서 다룹니다.

07 게이트웨이 만들기·관리

게이트웨이 만들기·관리

정책을 집행하려면 먼저 게이트웨이를 객체로 만들어 관리 서버와 신뢰를 맺어야 합니다. 이 장은 게이트웨이 객체를 만드는 법과, 운영 중 토폴로지·라이선스·핫픽스를 관리하는 법을 정리합니다.

새 게이트웨이 만들기

Gateways & Servers > **New** > **Gateway** 에서 **Classic Mode** 를 고르면 게이트웨이 속성 창이 열립니다. 흐름은 이름·IP 입력 → **Communication** 으로 신뢰 맺기 → 플랫폼(어플라이언스 모델) 선택 → One-time password 입력 → **Initialize** 로 SIC 신뢰 수립 → 토폴로지 가져오기 → 하드웨어·버전·OS 지정 → 켤 Software Blade 선택 입니다.

여기서 두 가지가 중요합니다. 플랫폼(어플라이언스 모델)을 정확히 골라야 — 틀리면 정책 설치가 실패할 수 있습니다. 그리고 신뢰의 바탕은 **SIC(Secure Internal Communication)** 로(인증 인프라), 관리 서버와 게이트웨이가 SSL로 서로를 인증 합니다. 일부 블레이드는 켤 때 first-time 마법사가 열리는데, 지금 또는 나중에 실행할 수 있습니다.

토폴로지 관리 — 수동과 동적

네트워크가 바뀌면 게이트웨이 토폴로지를 갱신 해야 합니다. **수동** 갱신은 게이트웨이 객체의 **Network Management** 에서 **Get Interfaces** 를 눌러 토폴로지까지 가져올지(Get Interfaces With Topology), 인터페이스만 가져올지 고르고, Anti-Spoofing 설정을 잡은 뒤 정책을 설치합니다(Bridge에 속한 물리 인터페이스는 토폴로지가 "Undefined"로 나오므로 API `get-interfaces` 사용). **동적** 갱신은 토폴로지 변화를 자동으로 반영 하게 해 줍니다.

라이선스·상태·핫픽스 관리

게이트웨이를 운영하면서 다루는 일들이 이 장에 모여 있습니다.

라이선스 관리(Managing Licenses) 와 **Security Gateway Indicators**(상태 지표)로 게이트웨이의 라이선스·건강 상태를 확인하고, **Central Deployment** 로 여러 게이트웨이에 핫픽스·버전 업그레이드를 중앙에서 일괄 배포 합니다. 또 게이트웨이가 관리 서버·Log Server에 접근하도록 구성 하거나, Implied Rule·커널 테이블을 조정 하며, **HealthCheck Point Tool** 로 종합 점검을 합니다.

이 모든 작업의 바탕에는 SIC로 맺은 신뢰 가 깔려 있어, 관리 서버가 안전하게 게이트웨이를 제어합니다. 게이트웨이에 무엇을 설치할지는 결국 정책 관리 기초와 Access Control 정책으로 이어집니다.

08 정책 관리 기초

정책 관리 기초

여러 종류의 정책을 하나로 묶어 함께 설치하고, 설치 이력을 관리 하는 것이 정책 관리의 핵심입니다. 이 장은 Policy Package 개념, 설치 과정, 그리고 설치 이력·동시 설치 같은 운영 기능을 정리합니다.

Policy Package — 정책을 묶는 단위

Policy Package 는 서로 다른 종류의 정책을 한데 묶은 것 으로, 설치하면 게이트웨이가 그 안의 모든 정책을 집행합니다. 한 패키지에는 다음을 담을 수 있습니다 — **Access Control**(Firewall·NAT·Application Control & URL Filtering·Content Awareness·Mobile Access), **QoS**, **Desktop Security**, **Threat Prevention**(IPS·Anti-Bot·Anti-Virus·Threat Emulation·Threat Extraction·Zero Phishing), **HTTPS Inspection**(R82부터 Inbound·Outbound로 나뉨).

핵심 발상은 사이트 유형별로 다른 패키지 입니다. 예를 들어 어떤 지점은 Firewall·VPN만, 어떤 곳은 QoS·Mobile Access까지 쓴다면, 각 사이트의 게이트웨이에 설치된 블레이드에 맞는 정책 조합을 패키지로 만들어 둡니다.

!사이트 유형별 정책 패키지 예 *① Sales California(Firewall·VPN) ② Sales Alaska(Firewall·VPN·IPS·DLP) ③ Executive management(Firewall·VPN·QoS·Mobile Access) ④ Server farm(Firewall) ⑤ 인터넷*

패키지는 **Manage policies and layers** > **New** 에서 만들며, General에서 정책 종류를 선택하고, Installation targets에서 설치할 게이트웨이(전체/특정)를 지정 합니다. 이렇게 패키지마다 적절한 설치 대상을 미리 묶어 두면 설치 때마다 게이트웨이를 다시 고를 필요가 없습니다.

정책 설치와 사용자 데이터베이스

Install Policy 를 누르면 설치가 시작됩니다. 설치 과정은 규칙을 휴리스틱 검증(일관성·중복 확인) 하고 — 검증 오류가 있으면 설치를 멈추고, 경고만 있으면 경고와 함께 설치 — , 각 게이트웨이가 최소 한 규칙은 집행하는지 확인(아니면 기본 drop 규칙 적용) 하며, 사용자·객체 데이터베이스를 설치 대상에 배포 합니다.

설치 모드는 두 가지입니다. **각 게이트웨이에 독립적으로 설치**(한 곳이 실패해도 나머지엔 영향 없음) 와 **모든 게이트웨이에 설치하되 한 곳이라도 실패하면 같은 버전 게이트웨이에는 설치 안 함** 입니다(클러스터는 "모든 멤버에 설치, 안 되면 전부 취소" 옵션도).

변경 내용에 따라 무엇을 설치할지 갈립니다. 정책 규칙 등을 바꿨으면 **정책 설치**, 사용자·관리자 정의만 바꿨으면 **Install Database** 입니다. 사용자 데이터베이스는 게이트웨이에는 정책 설치 때, 관리 블레이드를 켠 호스트에는 DB 설치 때 들어갑니다(CLI로는 `fwm dbload`). 한편 게이트웨이에서 `fw unloadlocal` 로 Access Control 정책을 내릴 수 있는데, 이 명령은 IP Forwarding을 꺼 모든 트래픽을 막고 모든 정책을 제거 하므로 주의해야 합니다.

설치 이력·동시 설치·가속

운영을 돕는 기능들이 있습니다. **Installation History** 로 누가 언제 무엇을 설치했는지 보고, 특정 버전으로 되돌려 마지막 "정상" 정책을 설치 할 수 있습니다. **Rule Logs** 로 특정 규칙이 만든 로그 를 보거나(Rule UID로 검색), 규칙의 변경 이력(History)을 봅니다.

성능 면에서, R81부터 **Concurrent Install Policy** 로 여러 게이트웨이에 서로 다른 정책 설치를 동시에(최대 5개, 초과분은 큐 대기, Access Control·Threat Prevention만 지원) 돌릴 수 있고, **Accelerated Install Policy** 로 마지막 설치 이후 변경분에 따라 Access Control 정책 설치 시간을 크게 단축 합니다(예: Host 객체를 만들어 규칙에 더하면 가속 설치 트리거).

09 Access Control 정책 만들기

Access Control 정책 만들기

이 가이드에서 가장 큰 장입니다. Check Point의 핵심인 하나로 통합된(unified) Access Control 정책 을 어떻게 짜는지 — 규칙의 구성, 매칭 방식, 레이어, 설치까지 — 의 큰 줄기를 잡습니다. 세부는 분량이 방대하니 원문 해당 절을 참고하세요.

통합 Access Control 정책

핵심은 여러 기능을 하나의 Rule Base로 통합 한다는 것입니다. Firewall(접근 통제), Application & URL Filtering(앱·사이트 차단), Content Awareness(데이터 타입 제한), IPsec VPN·Mobile Access(보안 통신), Identity Awareness(사용자·컴퓨터·네트워크 식별) 를 따로 관리하지 않고 한 규칙에 녹 입니다.

그래서 "특정 네트워크의 사용자가 특정 애플리케이션은 쓰되, 일정 크기 이상 파일 다운로드는 막는다" 같은 직관적 규칙 하나에 Security Zone·Service·Application·Data Type·Access Role 객체를 함께 쓸 수 있고, 그 결과가 하나의 로그(네트워크·프로토콜·앱·사용자·접근 자원·데이터 타입) 로 모입니다.

Rule Base의 열(컬럼)

규칙은 여러 열로 이뤄집니다. **No**(번호), **Hits**(매칭 횟수), **Name**, **Source/Destination**(트래픽의 시작·목적지 — Network·Host·Zone·Access Role·Updatable Object 등), **VPN**(적용 VPN Community), **Services & Applications**, **Content**(보호할 데이터 — 방향: Download/Upload/Any), **Action**, **Track**(로깅), **Install On**(규칙을 받을 게이트웨이), **Time**, **Comment** 입니다. 일부는 기본으로 숨겨져 있어 헤더 우클릭으로 켭니다.

Action 에는 Accept·Drop·Reject·Ask·Inform(UserCheck 메시지)·Inline Layer 가 있습니다.

규칙 매칭 — 가장 먼저 맞는 규칙

게이트웨이는 연결에 적용할 규칙을 위에서부터 찾 습니다(matching). 이 동작을 이해하면 Rule Base 성능을 끌어올리고 로그를 해석 하는 데 도움이 됩니다. 연결의 첫 패킷(SYN)에서 위에서부터 검사해 처음 맞는 규칙을 적용하고, 거기서 멈춰 나머지 규칙의 검사 엔진은 켜지 않습니다. 그래서 규칙 순서가 결과를 좌우 합니다.

Ordered Layer와 Inline Layer

규칙이 많아지면 **레이어** 로 정리합니다. **Ordered Layer** 는 Rule Base를 관리하기 쉬운 묶음으로 나누고 여러 패키지에서 재사용, **Inline Layer** 는 규칙 안의 독립된 하위 정책(sub-policy) 입니다. 덕분에 Rule Base를 평면이 아닌 계층 구조로 만들고, 레이어별로 관리자에게 소유권을 위임 할 수 있습니다.

Inline Layer는 부모 규칙(Action이 레이어 이름)과 하위 규칙 으로 이뤄집니다. 패킷이 부모 규칙에 안 맞으면 다음 Ordered Layer 규칙으로 넘어가 고, 부모에 맞으면 하위 규칙을 검사 합니다. 이때 Inline Layer 끝에는 항상 명시적 Cleanup Rule을 두고, 그 Action을 Implicit Cleanup Rule과 같게 하는 것이 권장됩니다(없으면 Implicit Cleanup Rule 적용).

정책 설치와 그 주변

정책을 다 짜면 검증을 거쳐 게이트웨이에 설치 합니다(정책 관리 기초). 이 장은 그 밖에 **Best Practices**(규칙 작성 모범 사례), **Hit Count 분석**(규칙 사용 빈도), **IP Spoofing 방지**, **NAT 정책 구성**, 그리고 Mobile Access·Site-to-Site VPN·Remote Access VPN 연동과 Implied Rules(암묵 규칙) 까지 폭넓게 다룹니다. 각 주제의 상세 절차는 원문 해당 절과 전용 가이드(Site-to-Site VPN·Remote Access VPN 관리자 가이드)를 참고하세요. 요지는 하나의 통합 Rule Base에 모든 접근 통제를 녹이고, 위에서부터 매칭되며, 레이어로 정리해 설치 한다는 한 줄기입니다.

10 Threat Prevention·UserCheck

Threat Prevention·UserCheck

이 장은 위협 방지 정책의 위치와, 사용자에게 직접 말을 거는 **UserCheck** 를 함께 다룹니다.

Threat Prevention 정책

관리 서버에서도 봇·바이러스 등을 검사하는 Threat Prevention 정책 을 만들지만, 그 자세한 구성은 별도의 R82 Threat Prevention 관리자 가이드 에서 다룹니다. 정책 관리 기초에서 봤듯 Threat Prevention은 IPS·Anti-Bot·Anti-Virus·Threat Emulation·Threat Extraction·Zero Phishing 을 포함하며, Policy Package에 담겨 게이트웨이에 설치됩니다.

UserCheck — 사용자에게 직접 알리기

UserCheck 를 켜면, 규칙에 따라 게이트웨이가 사용자에게 직접 메시지를 보내 위험하거나 규정에 어긋나는 행동을 알립니다. 이렇게 사용자가 스스로 보안 사고를 막고 조직 정책을 익히게 돕고, 기록된 사용자 응답을 바탕으로 정책을 다듬 을 수 있습니다. UserCheck 객체를 만들어 Rule Base에서 사용자와 소통합니다.

UserCheck를 지원하는 블레이드는 DLP, Access Control(Application Control·URL Filtering·Content Awareness), Threat Prevention(Anti-Bot·Anti-Virus·Threat Emulation·Threat Extraction·Zero Phishing) 입니다.

게이트웨이에서 UserCheck 구성

설정은 게이트웨이 객체의 **UserCheck** 페이지에서 합니다. 흐름은 이렇습니다.

먼저 **Enable UserCheck for active blades** 를 켭 니다. **UserCheck Web Portal** 의 Main URL(알림을 보여 주는 포털 주소) 을 확인하고(원격 접속 사용자는 게이트웨이 내부 인터페이스를 Main URL과 같게), 필요하면 **Aliases** 로 다른 호스트명을 Main URL로 리디렉션 합니다(회사 DNS에서 포털 IP로 해석되어야 함).

Certificate 항목에서는 포털이 관리 서버에 인증할 인증서를 import 합니다. 기본은 Check Point 내부 인증 기관(ICA)의 인증서 를 쓰는데(인증 인프라), 브라우저가 Check Point를 신뢰하지 않으면 경고가 뜰 수 있으니 공인 외부 기관의 인증서를 import 하면 경고를 막습니다. 마지막으로 **Accessibility** 에서 포털에 접근할 수 있는 인터페이스 를 토폴로지에 맞춰 구성합니다.

사용자에게 메시지를 띄우는 방식은 두 가지입니다 — 게이트웨이의 **UserCheck Web Portal** 로 리디렉션 하거나, 단말에 **UserCheck Client** 를 설치 하는 것입니다(Security Gateway 가이드의 UserCheck에서도 다룸).

11 HTTPS Inspection

HTTPS Inspection

HTTPS 트래픽은 TLS로 암호화되어 안전하지만, 그 안에 불법 행위나 악성 트래픽이 숨 을 수 있습니다. 게이트웨이는 암호화된 HTTPS를 그대로는 못 보므로, **HTTPS Inspection** 을 켜서 게이트웨이가 외부 서버와 새 TLS 연결을 맺어 복호화·검사 하게 합니다.

두 가지 검사 방향

Outbound HTTPS Inspection 은 내부 클라이언트가 외부 사이트로 보내는 트래픽 을 악성으로부터 보호하고, **Inbound HTTPS Inspection** 은 인터넷에서 내부 서버로 오는 악성 요청 으로부터 보호합니다.

이때 게이트웨이는 인증서를 써서 클라이언트와 보안 사이트 사이의 중개자 가 됩니다. 모든 데이터는 HTTPS Inspection 로그에 비공개로 보관 되며, HTTPS Inspection 권한이 있는 관리자만 로그의 모든 필드를 볼 수 있습니다.

Outbound 연결의 검사 흐름

나가는(Outbound) 연결은 내부 클라이언트에서 외부 서버로 가는 HTTPS 입니다. 흐름은 이렇습니다 — HTTPS 요청이 게이트웨이에 도착 → 게이트웨이가 검사 → HTTPS Inspection 규칙과 매칭되는지 확인 합니다. 규칙에 안 맞으면 HTTPS 페이로드는 검사하지 않 고, 맞으면 복호화해 검사 를 이어 갑니다.

즉 핵심은 게이트웨이가 새 TLS 연결로 중개하며, 규칙에 매칭될 때만 복호화해 들여다본다 는 것입니다. R82부터 HTTPS Inspection 정책은 Inbound Policy와 Outbound Policy로 나뉘 며(정책 관리 기초), 이 검사 결과는 Anti-Bot·Anti-Virus·Application Control·Content Awareness·DLP·IPS·Threat Emulation·URL Filtering 같은 블레이드가 활용합니다(Access Control 정책).

인증서 발급·관리는 내부 인증 기관(ICA)이 바탕이 되며(인증 인프라), 인증서를 전용 하드웨어에 보관하려면 Security Gateway 가이드의 HSM을 참고하세요. 규칙 구성·인증서 발급의 세부 절차는 원문 해당 절을 따릅니다.

12 외부 연동·IoT 보안

외부 연동·IoT 보안

관리 서버는 외부 데이터·클라우드·IoT 환경과 연결 해 정책의 폭을 넓힐 수 있습니다. 이 장은 External Network Feed, Infinity Portal 연동, 그리고 IoT 기기 보안을 묶어 정리합니다.

External Network Feed — 외부 데이터를 규칙에 쓰기

Network Feed 객체는 외부 HTTP/HTTPS 서버가 만드는 피드(IP·도메인 목록)를 규칙에 끌어다 쓰는 네트워크 객체입니다. 단일 IP·범위·`IP/masklen`·FQDN·non-FQDN 도메인을 담을 수 있고, 게이트웨이가 외부 소스의 변경에 맞춰 자동으로 가져와 갱신 하므로 정책을 다시 설치할 필요가 없 습니다. Access Control·HTTPS Inspection·NAT 정책에서 출발지·목적지로 씁니다.

이점은 수동 유지보수가 줄고, 정책 설치 횟수가 줄며, 구성이 단순해진 다는 것입니다. 다만 게이트웨이가 피드 서버에 HTTP/HTTPS로 접근할 수 있어야 하고, 한 게이트웨이는 Network Feed 객체 최대 500개(객체당 IP 최대 5만, 도메인 무제한) 를 지원하며, Dynamic·Updatable·Generic Data Center·Network Feed 객체를 합쳐 총 5,000개·IP 35만·도메인 12,500 까지입니다(지원 형식: flat list·JSON).

Infinity Portal 연동

온프레미스 관리 서버·게이트웨이를 Check Point의 클라우드인 Infinity Portal에 연결 할 수 있습니다. 그러면 Infinity Portal에서 관리되는 서비스를 내 서버·게이트웨이에서 실행 하고, 클라우드·온프레미스의 모든 Check Point 제품 로그를 한곳에서 통합 조회 하며, 온프레미스 관리 서버에서 새 관리자 기능(예: 어디서든 안전하게 관리 API 실행) 을 쓸 수 있습니다.

연결하려면 각 서비스에 유효한 라이선스 가 있어야 하고, 권한 프로파일에 **Manage integration with Infinity Services** 가 켜져 있어야 합니다. SmartConsole의 **Infinity Services** 뷰에서 **Get Started** → Infinity Portal에서 토큰을 받아 관리 서버와 신뢰를 맺으면 됩니다.

IoT 기기 보안

병원·산업·스마트빌딩의 IoT 기기(HVAC·프린터·엘리베이터·감시카메라·PLC·MRI 등)는 사이버 공격에 취약 합니다. 관리 서버는 이들을 위한 보안 정책을 제공합니다.

> 중 요
>
> R81.20 이상에서 Infinity Portal의 IoT Network Security 애플리케이션으로 IoT 정책을 구성 하면, SmartConsole의 IoT Policy Layer 객체는 읽기 전용 이 됩니다. 편집은 Infinity Portal의 IoT Network Security 앱에서만 합니다.

즉 IoT 보안은 Infinity Portal 연동과 맞물려 동작하며, 위의 Infinity 연결을 전제로 한 IoT 정책 관리가 핵심입니다. 세부 전제·절차는 원문 해당 절을 참고하세요.

13 관리 서버 운영

관리 서버 운영

관리 서버를 안정적으로 굴리는 데 필요한 환경설정(리비전), 고가용성, 규정 준수 세 가지를 묶어 다룹니다.

환경설정 — Database Revisions

Security Management 아키텍처에는 **Revision(리비전)** 이 내장되어 있습니다. publish할 때마다 직전과의 변경분만 담은 새 리비전이 생성 됩니다. 덕분에 위기 시 정상 리비전으로 안전하게 복구 하고, 설치 버전 간 차이만 보므로 정책 검증이 빠르며, Management 고가용성도 더 효율적 입니다.

다만 한계가 있습니다. Backup 관리 서버에서는 revert 불가, 이전 리비전으로 되돌리면 그보다 새 버전은 사라지는 비가역 작업, 객체에만 적용(파일 시스템·Task·SIC·License는 제외), revert는 다른 접속 사용자를 끊고 그들의 private 세션을 버림 입니다. 이 밖에도 SmartConsole의 **Preferences and Management Settings** 에서 여러 관리 설정을 다룹니다.

Management 고가용성(HA)

Management High Availability 는 관리 서버의 이중화·데이터베이스 백업 입니다. 동기화된 서버들은 같은 정책·규칙·사용자 정의·객체·시스템 설정 을 갖고, 내장 리비전 기술로 마지막 동기화 이후 변경분만 동기화 해 실시간 갱신을 최소 자원으로 해냅니다.

구성은 하나의 **Active** 서버와 하나 이상의 **Standby** 서버 입니다. 첫 설치 서버가 **Primary**(동기화 마스터), 이후가 **Secondary**(기본 Standby) 이며, Active가 일정 간격으로 ·publish할 때 Standby를 동기화합니다(미게시 세션은 동기화 안 됨). Standby는 읽기 전용으로만 열 수 있고, Active 장애 시 changeover는 자동이 아니라 수동 으로 합니다. Active와 통신이 끊겨 두 Active가 생긴 상태가 Collision Mode 인데, 이때는 동기화가 멈추며 한쪽을 Standby로 바꾸면 그 데이터가 Active 것으로 덮어쓰입니다.

Secondary 구성은 Primary의 SmartConsole에서 Secondary용 Check Point Host 객체를 만들고 Network Policy Management를 켠 뒤 SIC 신뢰를 맺고 publish 하면 동기화가 시작됩니다. 그다음 각 게이트웨이의 **Fetch Policy** 에 Secondary를 추가합니다. Primary가 영구히 못 쓰게 되면, `promote_util` 로 Secondary를 Primary로 승격 하고 원래 Primary IP로 새 Secondary를 세우는 재해 복구 절차를 따릅니다(라이선스는 IP에 묶이므로 승격 서버 IP로 재할당 필요).

Compliance — 규정 준수

Compliance 블레이드 는 Check Point 보안 인프라를 끊임없이 모니터링 하는 동적 솔루션입니다. **CCM(Continuous Compliance Monitoring)** 기술로 게이트웨이·블레이드·정책·설정을 방대한 규제 표준·보안 모범 사례 DB와 대조 하고, 문제를 바로잡을 시정 조치를 제안 합니다.

자동 스캔은 두 가지입니다 — 하루 한 번(CLI·스크립트로 바뀐 설정을 잡는) 일일 스캔 과 관리자가 게이트웨이·정책에 영향 주는 객체를 바꿔 publish하면 도는 스캔 (수동 스캔도 가능). 켜려면 관리 서버 객체의 **General Properties** > **Management** 에서 **Compliance** 를 선택 하고, 대시보드는 **Logs & Events** 뷰의 새 탭에서 **Compliance** 로 봅니다(Security Best Practices·Gateways 등 위젯 제공).

14 인증 인프라

인증 인프라

Check Point 환경의 신뢰는 내부 인증 기관(ICA)이 발급한 인증서 위에 섭니다. 이 장은 그 ICA를 다루는 도구, 모바일 기기용 클라이언트 인증서, 그리고 관리 서버를 통해 게이트웨이의 Gaia API를 실행하는 Gaia API Proxy를 묶어 정리합니다.

ICA 관리 도구

ICA(Internal Certificate Authority) 는 Management Server에 내장된 인증 기관 으로, SIC·VPN·사용자 인증서의 바탕입니다. **ICA Management Tool** 에서 관리자는 인증서 관리, CRL 재생성, ICA 파라미터 구성, 만료 인증서 제거 를 할 수 있습니다.

> 주 의
>
> ICA Management Tool로 SIC 인증서나 VPN 인증서를 바꾸지 마세요. SIC·VPN 인증서는 SmartConsole에서만 바꾸고, 이 도구는 사용자 인증서 작업(생성 등)에만 씁니다.

ICA는 인증서·CRL 모두 X.509 표준을 완전히 준수 합니다. 이 도구는 기본적으로 비활성 이라, 쓰려면 SmartConsole에서 관리자·사용자 객체와 인증서를 만들고, 관리 서버 CLI에서 `cpca_client set_mgmt_tool add …` 로 허용 사용자를 추가 해야 합니다(모범 사례는 sk102837).

모바일 기기용 클라이언트 인증서

스마트폰·태블릿 사용자가 클라이언트 인증서로 게이트웨이에 인증 하게 할 수 있습니다. 많은 조직에서 인증서 발급·유지는 게이트웨이 관리 부서와 다른 부서(예: 헬프데스크) 가 맡으므로, 인증서 발급만 허용하고 나머지 권한은 제한한 관리자 를 둘 수 있습니다(관리자·사용자 계정 관리).

발급은 SmartConsole의 **Security Policies** > **Access Control** > **Access Tools** > **Client Certificates** 에서 합니다. Check Point Mobile Apps는 인증서만으로, 또는 인증서 +사용자명/암호의 2단계 인증 을 쓰며, 인증서는 Mobile Access 게이트웨이를 관리하는 서버의 내부 CA(ICA)가 서명 합니다. 이 페이지에서 인증서를 만들고·편집·폐기하고, 상태·만료일·등록 키를 보고, 사용자에게 배포(이메일 템플릿) 합니다.

Gaia API Proxy

Gaia API Proxy 는 관리 서버를 거쳐 관리 대상 게이트웨이·클러스터 멤버의 Gaia API를 실행 하게 해 줍니다. 즉 API 클라이언트가 관리 서버에 접속하면, 거기서 각 게이트웨이의 Gaia API 명령을 돌릴 수 있습니다.

!Gaia API Proxy 구성 *① API 클라이언트 ② Gaia API Proxy 기능을 가진 Management Server ③ 관리 대상 Security Gateway ④ 관리 대상 ClusterXL · Ⓐ Management API 통신 Ⓑ Gaia API 통신*

흐름은 Management API `login` 명령으로 관리 서버에 로그인해 SID 토큰을 받고, 그 토큰을 Gaia API 명령의 `X-chkp-sid` 필드에 넣어 게이트웨이에 실행 하는 것입니다(로그인 관리자는 **Run One Time Script** 권한 필요). 단, Scalable Platforms(ElasticXL·Maestro·Chassis)는 이 기능을 지원하지 않 습니다.

```
POST https://<관리 서버 IP>/web_api/gaia-api/<버전>/<Gaia API 명령>
```

15 명령줄 참조

명령줄 참조

관리 서버를 깊이 다루다 보면 명령줄이 필요할 때가 옵니다. 원문에서 가장 방대한 부분(340여 페이지)이 바로 이 명령 참조인데, 모두 옮기기보다 어디에서 무엇을 찾는지 를 짚어 둡니다.

관리 서버 운영 명령 전체는 **R82 CLI Reference Guide** 에 정리되어 있습니다. 이 가이드 곳곳에서 본 명령들도 거기에서 자세히 다룹니다 — 예를 들어 사용자 데이터베이스를 설치하는 `fwm dbload` (정책 관리 기초), 게이트웨이에서 정책을 내리는 `fw unloadlocal`, API Server를 다루는 `api restart` · `api status` 와 `mgmt_cli` (API로 관리), 고가용성에서 Secondary를 승격하는 `promote_util` 과 백업용 `migrate export/import` · `mds_backup/mds_restore` (관리 서버 운영), ICA 도구를 여는 `cpca_client set_mgmt_tool` (인증 인프라) 등입니다.

특히 자동화의 중심인 `mgmt_cli` (Gaia·SmartConsole 포함)와 Web Services API(`/web_api/`) 의 전체 명령·옵션은 **Check Point Management API Reference** 에서 봅니다. Session 관리, Host·Network·Rule 명령 등을 다루며, SmartConsole의 API 창(F9)에서도 참조를 열 수 있습니다.

정리하면, 일상 관리는 SmartConsole로 하되 자동화·스크립팅·문제 해결이 필요할 때 CLI(`mgmt_cli` · `fwm` · `api` · `cpca_client` 등)와 관리 API 로 내려가며, 그 방대한 명령 사전은 CLI Reference Guide와 Management API Reference 가 담당합니다.