SOAR 구매자 가이드

본 가이드는 SOAR 솔루션을 구매할 때 필요한 핵심 평가 기준 및 고려사항을 다룹니다

splunk> turn data into doing™

목차

1. 소개 3

a. SOAR(보안 오케스트레이션 및 자동화, 대응) 정의 4

b. 보안 유즈케이스 식별 4

2. 평가기준 6

a. 핵심 기능 7

b. 플랫폼 속성 12

c. 비즈니스 고려사항 15

3. 결론 16

4. 평가 체크리스트 17

1. 소개

SOAR(Security Orchestration, Automation and Response) 플랫폼에 투자하는 것은 현명하고 매우 전략적인 결정입니다. 무엇보다 보안 운영 센터(SOC, Security Operations Center)를 구축하기 위한 플랫폼을 선택하는 것은 여타 개별 보안 제품을 선택하는 것보다 중요합니다.
SOAR 플랫폼은 보안 인프라의 중심축이 되어 보안 투자를 위한 운영 시스템 역할을 효과적으로 수행하기 때문입니다.

본 가이드는 SOAR 플랫폼을 평가할 때 고려해야 할 중요한 기준을 제시하는 것을 목표로 합니다.

“보안 오케스트레이션 및 자동화, 대응” 정의

자동화(automation)가 세일즈, 마케팅, 인사 및 IT 등 다양한 분야에서 각광을 받았지만, 보안 분야는 오케스트레이션과 자동화의 이점에 이제 막 눈을 떴습니다. 이러한 인식으로 인해 SOAR 플랫폼에 대한 구매자의 관심이 높아지고 있습니다. 그 결과 많은 보안 벤더가 SOAR 카테고리로 방향을 틀어서 인접 시장 영역에서 사용 중이던 자사의 기존 제품과 서비스를 이용해 인지도(mindshare)를 점유하려 하고 있습니다. SOAR 영역에 대한 기대가 커짐에 따라 서로 다른 관점을 가진 벤더의 수가 계속 증가했고, 결국 SOAR 시장의 정의가 흐려지고 제안하는 내용 간의 비교가 어려워지게 됐습니다.

따라서 다음과 같은 명확한 정의를 먼저 제시하고자 합니다.

보안 오케스트레이션(Orchestration)

보안 오케스트레이션은 기계(machine)를 사용하여 복잡한 인프라 전반에 걸쳐 이뤄지는 일련의 상호 의존적인 보안 액션(action)의 조정(coordination)을 의미합니다.

보안 자동화(Automation)

보안 자동화는 기계를 기반으로 한 보안 액션의 실행을 일컫습니다.

보안 대응(Response)

보안 대응은 이벤트, 케이스(티켓) 및 사고 워크플로우에 대한 사용자 및 기계 기반의 활동을 정책 기반으로 조정하는 것을 일컫습니다.

이 가이드에서 상기 정의를 사용하여 동급 최고의 SOAR 플랫폼을 구성하는 기능, 속성 및 고려 사항을 더 자세히 알아볼 예정입니다.

보안 유즈케이스의 식별

일반적으로 보안팀에서는 SOAR 플랫폼으로 어떤 보안 유즈케이스(use case)를 구현할지 식별합니다. 유즈케이스는 기존의 수동 워크플로우를 모델로 하며 일반적으로 운영상의 최대 페인 포인트(pain point)을 반영합니다. 워크플로우에는 많은 수동 태스크(task)가 포함되며, 이를 완료하려면 다수의 제품을 포괄하는 작업이 필요한 것이 일반적입니다.

평가를 시작하기 전에 페인 포인트 외에도 모든 잠재적 유즈케이스를 매핑하는 것도 중요합니다. 이러한 과정에는 보안팀 전체의 주요 이해 관계자가 참여해야 합니다. 즉시 구현되지 않더라도 포괄적인 유즈케이스를 규명하는 것은, 현재 선택한 플랫폼이 미래의 요구사항을 지원하는데 도움이 되기 때문입니다.

다음의 내용은 조사(investigation), 보강(enrichment), 격리(containment) 및 해결(remediation)에 대한 보안 유즈케이스를 선별한 것입니다.

경보 분류(Alert Triage)

경보 분류는 현재 발생한 경보를 검증하고 우선순위를 설정하는 것을 목적으로 합니다. 추가 컨텍스트(context)를 사용하여 이벤트를 보강하는 것이 수신된 경보 분류 유즈케이스에 해당됩니다. 또한 오탐(false positive) 경보 제거 로직이 여기에 포함될 수 있습니다.

사고 대응(Incident Response)

사고 대응 유즈케이스는 사고의 유형에 따라 크게 다를 수 있습니다. 예를 들어, 피싱 시도 사고에 대응하는 것과 성공적인 랜섬웨어 공격에 대응하는 것은 크게 다릅니다.

침해지표(IOC) 헌팅

IOC 헌팅을 자동화함으로써 보안팀은 리소스 제약으로 인해 헌팅 대상 IOC를 제한하는 일 없이, 수신한 위협 인텔리전스를 완전하게 활용할 수 있습니다.

또한 인텔리전스 스코어링을 구현하여 어떤 위협 인텔리전스 소스를 사용할지 결정하는 데 도움을 받을 수 있습니다.

취약점 관리(Vulnerability Management)

취약점을 식별, 분류, 개선 및 완화하는 사이클을 자동화하면 보안팀의 효율성이 향상될 뿐만 아니라, 프로세스가 매번 동일한 방식으로 수행되어 더욱 일관적인 결과를 얻을 수 있습니다.

네트워크 접근 제어(NAC, Network Access Control)

SOAR 플랫폼은 동적 액세스 제어 전략을 강화할 수 있습니다. 이전에는 NAC 의사 결정 로직의 일부가 아니었던 탐지 시스템을 통합하는 것을 일례로 들 수 있습니다.

사용자 관리(User Management)

정확하고 신속하며 체계적으로 사용자의 계정을 활성화 및 비활성화 할 수 있으면, 위협 행위자가 사용자 계정을 악의적으로 사용할 가능성을 차단할 수 있습니다.

모의침투 테스트(Penetration Testing)

자산 식별, 분류 및 타겟 우선 순위 지정과 같은 활동을 자동화할 수 있고, 이에 따라 모의침투 테스트팀의 생산성이 향상됩니다.

인텔리전스 공유(Intelligence Sharing)

인텔리전스 공유 주도권(initiative)이 있는 조직은 자동화 지원 플레이북을 통해 큰 이점을 얻을 수 있습니다. 또한 자동화 수준도 향상될 수 있습니다. 분석가의 생산성을 높이고 수동 프로세스보다 더 빠르게 시급성을 요하는 정보를 커뮤니티에 제공할 수 있기 때문입니다.

기타 유즈케이스(Other Use Cases)

다른 자동화 유즈케이스 사례는 이미 잘 알려진 시나리오입니다. 보안 운영팀이 자동으로 결정을 내리고 적절한 액션을 취하는 데 사용할 수 있는 기준을 코드화하는 것입니다.

2. 평가 기준

본 가이드는 SOAR 평가 기준을 핵심 기능, 플랫폼 속성, 비즈니스 고려 사항의 최소 세 가지 영역으로 나눌 것을 제안합니다. 핵심 기능은 일반적이고 본질적인 기능이며 쉽게 식별됩니다. 플랫폼 속성은 플랫폼 선택에 영향을 미치는 기준을 형성하는 아키텍처 상의 특성과 같은 요소로 핵심 기능 대비 쉽게 식별되지 않습니다. 비즈니스 고려 사항은 제품에서 제안하는 사항을 보완하며, 핵심 기술을 보강하기 위해 회사에서 제공하는 훈련 및 지원 등의 부가 가치 서비스를 포함합니다.

핵심 기능

핵심 기능은 SOAR 플랫폼의 기본 구성요소로 볼 수 있습니다. 본 보고서에서는 각 기능 또는 구성 요소를 열거하고 평가와 선택을 지원하기 위한 고려 사항을 제시할 것입니다.

오케스트레이터(Orchestrator)

오케스트레이터는 주어진 보안 시나리오와 관련된 모든 활동을 처음부터 끝까지 지시하고 감독해야 합니다. 모든 상황에서 오케스트레이터가 일관되게 예측 가능한 결과를 제공하고 가용한 리소스의 활용을 최적화하는 것이 중요합니다.

데이터 수집(Data Ingestion)

보안 오케스트레이션과 자동화는 보안 데이터 수집에서 시작합니다. 오케스트레이터는 보안 데이터를 모든 데이터 소스와 포맷에서 수집할 수 있어야 합니다. 플랫폼으로 푸시되는 데이터를 수신할 수 있어야 하며, 데이터 소스(data source)를 폴링(polling)하고, 데이터를 플랫폼으로 가져올 수 있어야 합니다. 비정형 데이터가 수집되는 경우 오케스트레이터는 사용자가 데이터 핸들러를 제공하여 데이터를 해석하고 SOAR 플랫폼이 해당 데이터를 사용할 수 있도록 해야 합니다. 또한 오케스트레이터는 다수의 소스에서 데이터를 수집할 수 있어야 하며 수집된 데이터를 논리적으로 분리된 상태로 유지할 수 있는 옵션이 있어야 합니다.

의사 결정(Decision Making)

사용자는 데이터 소스에 적용되는 자동화 플레이북을 선택할 수 있어야 합니다. 예를 들어 이메일 피싱 플레이북은 이메일 기반 수집 소스에 적용되고 멀웨어 조사 플레이북은 SIEM(경보 수집 원천)에 적용될 수 있어야 합니다. 이 의사 결정 단계는 이후 설명할 “경보 관리” 기능과 밀접한 관련이 있습니다.

태스크 실행(Task Execution)

일반적으로 대기열 상의 자동화 태스크를 적절하게 최적의 시점에 배치하고, 이를 실행할 수 있게 자동화 엔진으로 전달하는 것은 오케스트레이터의 역할입니다.

사용자의 감독(Human Supervision)

오케스트레이터는 기계 기반 자동화와 필수적인 사용자의 감독 사이에서 효과적으로 균형을 이루어야 합니다. 일반적으로 분석가를 필요로 하는 세 가지 시나리오가 있습니다.

(1) 타겟에 대한 보안 액션을 실행하기 위해 자산 소유자의 승인이 필요한 경우, (2) 보안팀이 사업 연속성과 균형을 이루는지 확인하기 위해 분석가의 검토가 필요한 경우, (3) 분석가가 코드화된 의사 결정 로직을 보강해야 하는 경우(에러가 발생한 경우 등).

데이터 관리(Data Management)

또한 오케스트레이터는 특정 액션의 출력 데이터가 적절하게 파싱, 정규화 및 구조화되어 향후 액션에서 사용할 수 있도록 해야 합니다. 오케스트레이터는 다른 리소스에 부담을 주지 않기 위해 필요 시 관련 데이터의 캐싱도 지원해야 합니다.

무중단 운영(Fault Tolerance)

SOAR 플랫폼은 자동화 플레이북을 실행하기 위해 많은 개별 제품 및 서비스와 정기적으로 상호 작용합니다. 오케스트레이터는 제품 및 서비스에 대한 가용성이 항상 보장되지는 않을 것이라는 점을 예상해야 합니다. 외부 서비스에 대한 액세스에 차질이 있거나 중단될 수 있습니다. 이러한 상황에서 오케스트레이터는 예상 가능하게 작동해야만 하며, 사전 구성된 대로 정상적으로 작업을 복구하고 재개해야 합니다

자동화 엔진(Automation Engine)

자동화 엔진은 오케스트레이터에서 생성하는 액션 또는 태스크를 수신하고 안정적으로 실행하는 SOAR 플랫폼의 핵심입니다. 자동화 태스크는 대부분 사용자의 개입 없이 독립적으로 실행되기 때문에 플랫폼의 확장성 및 추가 가능성과 같은 속성은 중요하게 고려해야 할 기준입니다.

성능 확장성(Scalability)

자동화 엔진의 수직 및 수평 확장성을 이해하는 것은 중요합니다. 사용자는 시간이 지남에 따라 더 많은 유즈케이스를 자동화하는 것이 일반적입니다. 그리고 추가 유즈케이스마다 자동화 엔진에 추가적인 처리 부하가 발생합니다. 자동화 엔진은 수직적 확장성(CPU 및 RAM 리소스 증가 등)과 수평적 확장성(서버 인스턴스 증가 등)을 모두 고려하여, 성능을 향상하고 자동화에 대한 투자 효과(ROI)를 달성할 수 있도록 설계되어야 합니다.

유연성(Extensibility)

보안은 빠르게 발전하므로 대대적인 재설계 없이 자동화 엔진에서 새로운 기능을 지원할 수 있어야 합니다. 자동화 엔진은 주어진 환경에서 요하는 고유한 기능에 적응할 수 있는 기능을 지원해야만 합니다.

경보 관리(Alert Management)

데이터 수집 직후에 SOAR 플랫폼의 경보 관리 기능은 수신된 경보를 대기열에 넣고 이들의 우선 순위를 지정해서 분석가가 분류를 보다 효율적으로 수행할 수 있도록 해야 합니다. 경보에 대한 조사는 수동 또는 자동 액션을 수행할 수 있으며, 최고 수준의 분류 생산성 및 정확성을 산출할 수 있어야 합니다. 경보 관리 기능의 화면(UI)은 보안 경보의 모든 측면을 신속하게 확인하고, 효율적인 액션이 이뤄질 수 있도록 구축되어야 합니다. 또한 화면(UI)은 적시에 올바른 정보를 표시할 수 있도록 정보를 정렬해야 합니다.

경보 상세 정보(Alert Details)

보안 경보의 기술적 속성을 빠르게 요약하여, 분석가가 전반적 보안 시나리오를 이해할 수 있게 수 있도록 구성해야 합니다. 달리 말해 IP 주소, 도메인 이름, 파일 해시, 사용자 이름, 이메일 주소 및 기타 모든 관련 데이터 필드 등 정보를 유기적으로 보여주어야 합니다. 공통 이벤트 포맷(CEF)또는 이에 상응하는 표준 형식을 사용하는 것은 데이터 교환에 매우 유용합니다.

액션 발생(Issuing Actions)

경보를 조사할 때 보안 분석가는 경보 데이터를 사용하는 플랫폼에 수동 액션을 실행할 수 있어야 합니다. 여기에는 검사, 격리, 교정 또는 일반적인 액션 등이 포함됩니다. 화면(UI)은 사용자가 작업할 데이터를 선택하여 액션을 실행할 수 있도록 구성되어야 합니다. 이러한 행위를 컨텍스트를 고려한 액션 실행이라고도 하며, 새로 발견된 정보를 중심으로 분석을 확장해 나가는데 도움이 됩니다.

수동 액션 실행과 마찬가지로 분석가는 경보에 대한 일련의 액션을 생성 및 실행할 수 있어야 합니다. 이러한 액션 사항을 모아둔 것을 일반적으로 플레이북이라고 합니다.

액션의 결과(Action Results)

경보에 대해 수동 또는 자동화된 액션을 취하면, 해당 결과를 분석가에게 보여주고 분석가가 이해할 수 있어야 합니다. 뿐만 아니라 SOAR 플랫폼이 해당 내용을 이해하여 자동화된 결정을 하는데 도움을 줄 수 있도록 구성되어야 합니다. 액션의 결과는 요약 포맷(표 형식 등)과 보다 포괄적인 포맷(JSON 등)으로 제공되어야 합니다.

활동 로그(Activity Log)

플랫폼은 경보에 대해 실행된 모든 액션의 기록(수동으로 또는 자동화 플레이북으로 실행되었는지 여부와 무관하게)을 포함하는 포괄적인 활동 로그를 제공해야 합니다. 각 액션에 대한 성공 또는 실패에 대한 지표 등 결과를 표시해서, 해당 액션이 완전히 실행되었는지 여부를 명확히 보여주어야 합니다.

경보의 상태, 심각도, 민감도(Alert Status, Severity and Sensitivity)

플랫폼에서 관리하는 모든 경보에는 상태 지표(신규, 진행중 또는 완료), 심각도 지표 및 민감도 지표(신호 등 색상 표현 등)가 포함되어야 합니다. 각 지표는 경보 관리 화면(UI)과 플레이북 내에서 수정할 수 있어야 합니다.

경보 협업(Alert Collaboration)

분석가가 경보에 대해 협업하고, 의견을 남기고, 기타 정보를 제공할 수 있는 영역을 화면(UI)에서 제공해야 합니다. 이러한 협업 기록을 캡처하고 다른 모든 경보 데이터와 함께 제공이 가능하도록 구성하는 것이 이상적입니다.

케이스 관리(Case Management)

경보 또는 이벤트가 확인되고 에스컬레이션 되면 케이스 관리 구성 요소는 다양한 영역의 수명 주기를 포괄적으로, 생성부터 해결까지 보여줄 수 있어야 합니다. 케이스 관리에는 경보와는 구별되는 케이스 만의 추가 속성을 포함해야 합니다. 그 이유는 다수의 경보가 확인, 집계 및 에스컬레이션 과정을 거쳐 하나의 케이스가 되기도 하기 때문입니다. 또한 경보 관리는 일반적으로 기술적인 반면, 케이스 관리는 일반적으로 기술 및 비기술적 단계가 프로세스에 포함됩니다. 마지막으로 케이스는 경보에 비해 양이 적은 경향이 있습니다. 많은 조직에서 하루에 수백 또는 수천 건의 경보를 수신하는 반면, 케이스는 하루에 열 건 미만이 발생합니다.

케이스 데이터 구성(Case Data Organization)

케이스와 관련된 모든 데이터는 케이스 관리 구성 요소에 의해 집계되어야 합니다. 정보를 단일 위치에 표시하면 사용자가 정보를 효율적으로 확인할 수 있고, 컨텍스트 전환을 피할 수 있습니다

케이스에 데이터 추가(Adding Data to a Case)

케이스 관리 화면(UI)은 경보의 소스 데이터 및 액션 결과와 같은 관련 기술적 데이터를 케이스에 첨부하는 것을 지원해야 합니다. 화면(UI)은 또한 노트, 메모, 이메일, 스크린샷, 기록물 또는 케이스와 관련된 기타 임의 파일과 같은 비기술적 데이터가 첨부될 수 있게 지원해야 합니다. 케이스에 대한 정보의 자동 첨부도 플레이북 내에서 가능해야 합니다.

경보에 케이스 연결(Linking Cases to Alerts)

케이스 조사 중에는 추가 검사가 필요한 데이터나 즉각적인 격리 조치가 필요한 시나리오를 식별하게 되는 경우가 흔합니다. 따라서 분석가가 액션을 취해야 한다고 결정하는 경우, 케이스 관리 화면(UI)은 각 경보를 경보 관리 화면(UI)에 원활하게 연결할 수 있어야 합니다. 또한 경보 관리 화면(UI)에서 추가 액션을 실행할 수 있으며 관련 데이터에 대한 변경 사항은 케이스 관리 화면(UI)에 반영되어야 합니다

기존 프로세스에 매핑(Mapping to Existing Processes)

많은 조직에서 사고 대응, 비상 사태, 재해 및 기타 중요한 상황에 대한 표준 운영 절차(SOP)를 개발했습니다. 다음 단계는 케이스 관리 기능을 통해 사용자에게 자체적 프로세스에 따라 각 단계를 정의하고 템플릿의 형태로 저장할 수 있는 기능을 제공하는 것입니다. 사용자는 여러 단계로 SOP를 나눌 수 있어야 하고, 각 단계에 하나 이상의 태스크가 있고 각 태스크에 소유자를 할당할 수 있어야 합니다. 태스크와 관련된 추가적인 컨텍스트 정보가 태스크 설명에 포함된다면 더욱 좋습니다. 태스크 관리 애플리케이션과 마찬가지로, 태스크 소유자가 태스크를 완료했을 때 '완료'로 표시되어야 합니다. 화면(UI)은 케이스의 진행 상황과 케이스 상태에 대한 지표를 제공해야 합니다.

활동 감사(Activity Auditing)

정보 추가 또는 수정 및 상태 변경은 케이스 관리에 중요한 세부 정보입니다.

케이스에 대한 모든 변경 사항은 감사 추적에 기록되어야 하고, 외부로 내보낼 수 있어야 합니다. 케이스 변경 사항에는 다음이 포함될 수 있습니다.

- 데이터 추가
- 데이터 수정
- 단계 또는 태스크 수정
- 파일 또는 노트 추가
- 파일 또는 노트 수정
- 태스크 완료
- 케이스에 대한 기타 모든 활동 또는 수정

플레이북 관리(Playbook Management)

이상적인 모습이라면, 조직 내에서 그리고 잠재적으로는 커뮤니티 전체에 플레이북의 형태로 SOP를 관리하는 기능과 개정/수정 제어 기능을 제공할 수 있어야 합니다.

플레이북 구성(Playbook Organization)

플레이북 관리는 플레이북의 적절한 구성 및 그룹화를 지원해야 합니다. 사용자는 조직에 가장 적합한 내용에 따라 자신이 사용하는 그룹화를 정의할 수 있어야 합니다. 예를 들어 테마, 민감도, 조직 유형 또는 자산 유형에 따라 선택하여 플레이북을 구성 및 그룹화 할 수 있어야 합니다.

커스텀 함수(Custom Functions)

사용자가 즉시 사용할 수 있는 기능적 액션 외에도 커스터마이징과 확장성이 플레이북 생성에서 지원되어야 효과적인 보안 프로세스 자동화가 가능합니다. 사용자는 플레이북 내에서 커스텀 코드 블록 또는 커스텀 기능을 작성할 수 있어야 합니다. 이러한 커스텀 기능은 중앙 집중식 코드 관리 및 버전 제어를 제공하면서도, 다수의 플레이북에서 쉽게 공유할 수 있어야 합니다. 그렇게 된다면 사용자의 파이썬 코딩 능력 여부에 관계없이 플레이북 생성 및 실행이 크게 개선될 수 있습니다.

플레이북 대량 편집(Bulk Edits to Playbooks)

각 플레이북의 내부 작동 방식은 고유할 수 있습니다. 그러나 관리 관점에서는 많은 플레이북이 공통점을 갖고 있습니다.

따라서 플레이북 관리 시스템은 다음과 같은 플레이북의 대량 편집을 지원해야 합니다.

- 수집 소스
- 자동 실행 활성화/비활성화, 안전 모드 작동 활성화/비활성화
- 향상된 로깅 활성화/비활성화
- 플레이북 카테고리 그룹화 설정

형상 관리 및 배포(Revision Control and Distribution)

Git 등 버전 제어 시스템(VCS)과의 통합은 대규모 플레이북의 성공적 관리를 위한 필수적인 요소입니다. 배포 단계에서 VCS를 활용하면 다수의 시스템에 플레이북을 체계적으로 배포할 수 있습니다. 이러한 기능은 플레이북을 개발 시스템과 운영(production) 시스템을 동기화하거나 다수의 사이트에 걸쳐 있는 다수의 운영 시스템에서 동기화하는 데 유용합니다. 개발 단계에서는 VCS는 개정 및 변경 사항을 추적하고, 필요한 경우 변경 사항 원상복구를 가능하게 하는 요소입니다. 두 번째 이점은 개발자가 선택한 에디터에서 플레이북을 편집하고 수정된 플레이북을 플랫폼에 다시 쉽게 동기화할 수 있다는 것입니다.

자동화 에디터(Automation Editor)

자동화 에디터는 분석가 또는 관리자가 프로세스를 자동화된 플레이북으로 코딩하는 도구입니다. 비주얼 자동화 에디터의 전신은 기본 소스 코드 에디터입니다. 소스 코드 에디터에서만 자동화된 플레이북을 편집할 경우, 플레이북 구성은 소수의 개발자 집단만이 수행할 수 있는 지루하고 어려운 작업이 됩니다. 하지만 비주얼 자동화 에디터를 사용하면 소스 코드에서 플레이북을 작성할 능력이 없는 보안 전문가도 포괄적이고 정교한 플레이북 작성이 가능합니다. 비주얼 에디터는 비즈니스 프로세스 모델링 표기법(BPMN)을 준수해야 합니다. BPMN은 비즈니스 프로세스를 상세 표기하는 그래프 형태의 표기법입니다. BPMN은 비즈니스 사용자에게 직관적인 기호를 지원하는 동시에 기술적 지식이 있는 사용자에게 매우 복잡한 프로세스를 표현할 수 있는 기능을 제공합니다

사용자 화면(UI) 요소

사용자 화면(UI) 요소는 비주얼 플레이북을 구성할 수 있는 캔버스에서 시작해야 합니다. 또한 화면(UI) 상의 한 부분에서 사용자들이 원하는 액션을 명시할 수 있는 영역을 제공해야 합니다(예, block_ip 또는 file_reputation 등).

해당 액션이 선택되면 조치를 올바르게 구성하는 데 필요한 매개변수가 있을 수 있습니다. 따라서 화면(UI)은 매개변수를 수동으로 입력하거나 목록에서 매개변수를 선택할 수 있는 기능을 제공해야 합니다. 경보 데이터 및/또는 액션 결과 데이터도 매개변수로 사용될 수 있습니다.

또한 화면(UI)에는 테스트 및 디버깅이 수행될 수 있는 위치가 있어서 편집 모드에서 테스트 모드로 원활하게 전환할 수 있어야 합니다. 마지막으로 사용자가 자동화된 플레이북의 소스 코드를 보고자 하는 경우 소스 코드 표시 화면에 접근할 수 있어야 합니다.

코드의 블록 기반 표현(Block-Based Representation of Code)

블록을 사용하여 자동화 플랫폼에서 유의미한 단계를 표현할 수 있으며, 소스 코드를 건드리지 않고도 사용자가 포괄적이고 복잡한 플레이북을 작성할 수 있습니다. 실행 순서를 지정하기 위해 블록은 일대일, 일대다 또는 다대일 방식으로 연결되어야 합니다 구체적으로는 액션 실행, 플랫폼 API 호출, 조건문(if/then) 및 하나의 플레이북을 다른 플레이북에 연결하는 분기문 등을 포함하는 플레이북을 사용자가 비주얼 요소를 통해 구축할 수 있어야 합니다.

의사 결정 과정에 사용자 개입(Inserting Humans Into the Decision Process)

감독형 자동화 지원은 일반적인 요구 사항이 되었으며, 이는 플레이북 실행이 지속되는 상황에서 플레이북의 실행 승인, 검토 또는 강화를 위해 자동화 시퀀스에 사용자 개입을 포함하는 것을 의미합니다. 자동화 에디터는 하나 이상의 보안 액션과 인접한 플레이북에 승인 지점(들)을 삽입하여 사용자의 관리 감독을 지원할 수 있어야 합니다. 플레이북 작성자는 원하는 알림 또는 승인 유형을 지정하고, 또한 자동화 루프에 포함할 담당자를 지정할 수 있어야 합니다. 하나 이상의 평판 서비스를 사용할 수 없는 등의 상황에 대비하여, 플레이북 에디터와 플랫폼에는 사용자를 자동화 루프에 포함하는 에러 처리 로직을 정의할 수 있어야 합니다.

액션 결과의 정보 교환(Information Exchange of Action Results)

자동화 에디터 화면(UI)은 선행 액션 실행으로 인한 새로운 정보를 후행 액션 또는 결정 블록에 대한 입력 값 또는 매개변수로 사용할 수 있도록 허용해야 합니다. 선행 액션의 결과는 시각적으로 접근할 수 있어야 하며 업스트림 액션의 매개변수를 채울 때 드롭다운 목록에서 선택할 수 있어야 합니다.

플레이북 소스 코드에 대한 접근(Access to Playbook Source Code)

비주얼 에디터에서 플레이북을 구성한 결과 생성된 플레이북 소스 코드는 실시간으로 생성되고 작성자가 접근할 수 있어야 합니다. 물론 전통적인 소스 코드 도구를 통해 플레이북의 전체 또는 일부를 구성하는 것을 선호하는 사용자가 일부 있습니다. 이 경우 화면(UI)에서는 비주얼 에디터 대신 소스 코드 에디터로 대체할 수 있는 기능을 제공해야 합니다. 또한 비주얼 에디터 모드와 소스 코드 모드 간의 전환은 매우 원활하고 수월해야 합니다.

시각적 및 비시각적 플레이북의 동시 구축 (Simultaneous Visual and Non-Visual Playbook Construction)

플레이북의 소스 코드로 작업하는 경우, 자동화 에디터는 작성자가 소스 코드 수준에서 플레이북을 수정할 수 있도록 허용하는 동시에 비주얼 블록 수준에서 플레이북을 수정할 수 있는 기능을 제공해야 합니다. 또한 작성자가 비주얼 에디터의 범위를 벗어나는 커스터마이징을 위해 소스 코드 수준에서 수정해야 하는 개별 블록(액션, 결정 블록 등)이 필요한 경우가 있습니다. 이러한 수정이 완료된 후에도 사용자는 여전히 플레이북을 자유롭게 시각 도구를 사용하여 수정할 수 있어야 합니다.

내장형 테스트와 디버깅 및 런타임 로깅(Built-In Testing and Debugging and Runtime Logging)

통합 개발 환경(IDE)에서 실행 및 디버깅 기능을 제공하는 것은 표준사항에 해당합니다. 자동화 플랫폼을 사용하는 경우, 사용자는 보안 경보에 대해 플레이북을 실행하고 실행 활동 및 결과를 관찰할 수 있어야 합니다. 작성자가 소스 코드를 선호하는 경우 로깅 및 에러 코드는 비주얼 블록 에디터 또는 소스 코드 에디터와 동시에 표시할 수 있는 디버그 창에 표시되어야 합니다. 여기서 중요한 것은 작성자가 하나의 화면(UI) 내에서 플레이북을 빠르게 편집, 테스트 및 디버그 할 수 있도록 하는 것입니다.

안전 모드(Safe mode)

운영 이전 단계의 테스트가 필요한 신규 플레이북에 대해서는 안전 모드가 자동화 에디터에서 지원되어야 합니다. 안전 모드에서는 자동화 타겟에 대한 신규 플레이북 실행 시뮬레이션이 가능하지만, 변경 사항이 타겟에 영향을 주지는 않아야 합니다.

시뮬레이션 이후 작성자 또는 다른 플랫폼 사용자가 플레이북의 로직에 대해 충분한 확신을 갖게 되면, 안전 모드를 비활성화하고 플레이북을 정상적인 방식으로 기능하도록 설정이 가능해야 합니다.

앱 프레임워크(App Framework)

앱 프레임워크는 추가적인 화면(UI)을 제공하여, 현재의 보안 시장에 출시된 수많은 개별 제품들과 플랫폼을 연결하는데 도움을 줍니다.

개방형 생태계(Open ecosystem)

시장에서 제공되는 새로운 제품 및 서비스와 통합이 불가능하다면, SOAR 플랫폼은 시간이 지남에 따라 가치를 잃을 것입니다. 따라서 앞으로 출시될 앱을 안정적으로 통합하기 위해서는, 누구나 통합성에 대한 개발이 가능한 개방형 생태계를 선택해야 합니다. 이를 통해 사용자의 자립성을 강화함과 동시에 벤더 잠금 효과를 피할 수 있습니다. 또한 다양한 기술을 자동화된 플레이북에 추가 및 제거가 가능하며, 이 과정에서 플레이북에 부정적인 영향을 미치지 않습니다. 여기서 중요한 것은 핵심 플랫폼을 수정하지 않고, 새로운 기술이 플랫폼에 신속하게 통합이 가능해야 합니다. 마지막으로 추가 개발을 위해 SOAR 벤더에만 의존하지 않고, 사용자가 추가적 플랫폼에 대한 지원 방식을 통제할 수 있어야 합니다.

메트릭스 및 보고(Metrics & Reporting)

메트릭스 항목과 보고는 모든 자동화 플랫폼에 있어 중요하며 SOAR 플랫폼도 예외는 아닙니다. 자동화는 생산성 향상과 품질 향상을 약속합니다. 그리고 메트릭스는 자동화 플랫폼의 효율성을 이해하고 ROI 개선 지점을 식별하는 데 중요합니다.

유연한 대시보드(Flexible Dashboards)

각 조직과 개인 마다 다른 메트릭스를 사용합니다. 이 때문에 사용자는 자신의 조직에 가장 적합한 방식으로 메트릭스 항목을 구성할 수 있는 기능이 필요합니다. 따라서 SOAR 플랫폼은 고도로 맞춤화 된 방식으로 메트릭스 항목 정보를 구성할 수 있어야 합니다. 구체적으로는 대시보드 상에 어떤 메트릭스 항목이 어느 시간대에 표시될 것인지를 상세하게 보여줄 수 있는 정보 배치 순서의 구성도 포함됩니다.

성과 보고(Performance Reporting)

자동화를 사용하는 목적은 운영 효율성 증대입니다. 따라서 자동화가 제공하는 정량적 성과 향상 및 리소스 절감을 파악하고, 대시보드를 통해 해당 정보를 즉시 사용할 수 있도록 하는 것이 중요합니다.

플랫폼에서 지원해야 하는 주요 성과 측정 항목의 예는 다음과 같습니다.

- 평균 해결 시간(MTTR)
- 평균 체류 시간(MDT), (위협 행위자에 의한) 침해와 적절한 대응 사이의 시간으로 정의.
- 자동화된 실행을 통해 절약된 분석가의 시간
- 자동화된 실행을 통해 얻은 풀타임 환산 인력(FTE)의 수치
- 플레이북 실행 당 평균 절약 시간
- 절약된 비용(FTE 비용 x FTE 획득)

보안 효율성 보고(Security Effectiveness Reporting)

자동화를 사용하는 또 다른 이유는 조직의 보안 효율성 및 준비 태세 강화입니다. 현재 관리하고 있는 보안 경보의 총 개수와 관리 속도를 파악하는 것은 자동화가 제공하는 효과를 이해하는 데 있어 중요합니다.

플랫폼에서 제공해야 하는 주요 보안 효율성 메트릭스 항목은 다음과 같습니다.

- MTTR 및 MDT(위에서 소개됨)
- 진행 중 경보의 총 개수
- 하루에 개시된 경보(또는 시간, 주, 월 단위도 가능)
- 하루에 완료로 표시된 경보(또는 시간, 주, 월 단위도 가능)
- 서비스 수준 협약(SLA) 대비 성과

앱 통합 및 플레이북 성능(App Integration and Playbook Performance)

가장 자주 호출되는 플레이북을 이해하면 추가적으로 자동화 투자가 필요한 부분을 알 수 있습니다. 플레이북 설계 상에서 오탐 그리고 높은 신뢰도의 정탐 경보가 자동으로 처리되도록 구성하는 것이 이상적입니다. 자동화를 통해 경보 분류 상의 격차를 해결하지 못한다면 플레이북 개정이 필요할 수 있습니다

자동화 및 통합 효율성의 문제를 식별하려면 자동화 플랫폼에서 다음과 같은 메트릭스 항목을 제공해야 합니다.

- 자동화를 통해 완료로 표시된 경보(시간, 일, 주, 월 또는 기타 시간대 기준)
- 가장 많이 사용된 앱 통합
- 가장 많이 사용된 액션(수동 및 자동)
- 가장 많이 사용된 자동화 플레이북
- 플레이북 실행 시간
- 액션 실행 시간

인적 작업량(Human Workload)

자동화는 인적 자원 상의 격차를 줄이는 것을 목적으로 하지만 여전히 SOAR 플랫폼의 일상적인 활동에 사용자가 관여해야 하는 경우가 있습니다. 경보 상에서 수동 분류 및 기타 액션이 필요한 경우 또는 "감독형 자동화"를 달성하기 위해 사용자의 승인이 플레이북에 포함되는 경우 등이 대표적인 사례입니다. 이러한 인적 작업량을 이해하면 추가 자동화 및 조정이 필요한 분야 식별에 도움이 될 수 있습니다. 따라서 자동화 플랫폼에서 다음과 같은 메트릭스 항목을 제공하면, 자동화 프로세스와 관련된 인적 작업량 이해에 도움이 됩니다.

- 개인에게 할당된 경보
- 개인이 완료로 표시한 경보
- 평균 승인 시간
- 미결 승인 수
- 요청된 승인(시간, 일, 주, 월 또는 기타 시간대 기준)

플랫폼 속성

앞서 언급했듯 플랫폼 속성은 본질적으로 정성적 평가를 요합니다. 이러한 특징을 고려할 때, 플랫폼 속성은 플랫폼 상의 관찰과 상호 작용을 통해 평가하는 것이 일반적입니다.

배포 옵션(Deployment Options)

SOAR 플랫폼은 온프레미스, 클라우드 또는 하이브리드 배포를 지원해야 합니다. 물론 일부 보안 실무자는 온프레미스 배포를 선호하지만 다른 보안 실무자는 클라우드 배포를 선호하기도 합니다. 따라서 비즈니스 요구 사항을 가장 잘 지원하고 보안 운영을 간소화하며 디지털 전환을 촉진하는 방식으로 SOAR를 배포해야 합니다.

커뮤니티 기반(Community-Powered)

SOAR 플랫폼이 보안 운영 커뮤니티와 집중적으로 협업하는 것은 장기적인 성공과 직결될 수 있습니다. 그 이유는 한 업체가 모든 요소를 통합하기엔 시장에 너무 많은 제품, 서비스 및 플랫폼이 존재하기 때문입니다. 또한 진화를 거듭하는 보안의 특성상 최신 위협에 대처하기 위한 플레이북, 모범 사례, 전략을 공유할 수 있는 전문 커뮤니티와의 협업은 중요합니다. 따라서 SOAR 플랫폼은 굳건한 커뮤니티 모델을 지원하고 앱 통합과 플레이북을 쉽게 공유할 수 있어야 합니다.

대규모의 활동적인 사용자 커뮤니티(Large and Active Community)

커뮤니티와의 협업 잠재력을 측정하기 위한 방안 중 하나는 플랫폼의 설치 기반을 측정해 보는 것입니다. 대부분의 사용자는 비슷한 생각을 가진 다른 사용자의 경험을 활용하는 것을 선호합니다. 따라서 대규모의 활동적인 사용자 커뮤니티가 있다면 플레이북 또는 앱을 공유하거나, 새로운 자동화 유즈케이스에 대한 아이디어를 브레인스토밍 할 수 있는 기회가 생겨날 수 있습니다. 또한 커뮤니티에 벤더가 얼마나 적극적으로 참여하는지를 보면, 벤더가 커뮤니티와의 협업에 대해서 얼마나 진지한지를 알 수 있습니다.

상호간의 아이디어 교류를 용이하게 하려면 커뮤니티 내의 사용자들을 서로 연결되게 하는 것이 중요합니다. 일반적으로 그룹 및 다이렉트 메시징이 가능한 슬랙과 같은 커뮤니티 커뮤니케이션 도구를 제공할 경우 연결성은 높아집니다. 메시징 도구는 기술 및 설계 지원, 질문에 대한 빠른 답변 및 자동화 유즈케이스에 대한 브레인스토밍에 효과적이기 때문입니다. 새로운 아이디어를 전파하기 위한 추가 커뮤니케이션 도구로는 개인이 자신의 작업을 게시하는 커뮤니티 사용자 Github와 사용자 프레젠테이션, 커뮤니티 플레이북 및 사용자들이 직접 작성한 앱 통합 요소를 호스팅하는 중앙 집중식 커뮤니티 저장소 등도 있습니다.

협업(Collaborative)

협업이 이뤄진다면 플랫폼의 구성도 더욱 알차게 발전할 수 있습니다. 예를 들어 기능 완성도, 앱 통합의 커버리지, 진화하는 시나리오 범위를 다루는 자동화된 플레이북 등의 협업이 대표적 사례입니다.

커뮤니티 전반의 협업(Collaboration Across the Community)

콘텐츠 관점에서 보자면, 사용자와 벤더가 제공한 콘텐츠는 커뮤니티의 모든 사용자가 접근할 수 있는 중앙 저장소에 존재해야 합니다. 여기에는 플레이북 및 앱 통합과 같은 기술적 콘텐츠뿐만 아니라 프레젠테이션, 기술 노트, 블로그 및 기타 문서화 도구와 같은 비기술적 콘텐츠도 포함됩니다. 또한 정보 저장소는 커뮤니티의 규모를 반영하여 지속적으로 성장해야 합니다.

플랫폼 전반의 협업(Collaboration Across the Platform)

플랫폼이 제대로 협업을 이끌어내지 않는다면 협업이라는 퍼즐은 완성되지 않습니다. 사용자가 다양한 신뢰의 계층에서 발생하는 협업 내용을 활용할 수 있도록 하는 것은 SOAR 플랫폼의 주요 요건 중에 하나입니다. 플랫폼은 반드시 가장 민감한 협업, 즉 조직의 보안팀 전반에 걸친 커뮤니케이션을 지원해야 합니다. 여기에는 내장형 채팅 기능과 경보 및 케이스에 대한 노트 및 관련 데이터 첨부 기능 등도 포함됩니다.

인지(Cognitive)

인지적 SOAR 플랫폼이라고 하면 사람의 지식과 이전의 관측 사항을 활용하여 미래의 의사 결정에 도움이 될 수 있어야 합니다. 하지만 사람의 지식을 플레이북 형태로 자동화하기 위해서는 먼저 시스템에 코드화가 필요 합니다. 또한 기존의 실행 관련 통계, 수집된 데이터의 특성 및 액션 결과 등 과거의 관찰 내용을 활용할 수 있어야 합니다. 이러한 정보를 사용하면 개별 액션, 플레이북 뿐만 아니라 플레이북을 구성하게 되는 일련의 액션을 순서대로 추천이 가능합니다. 따라서 SOAR 플랫폼의 현재 인지 능력과 향후 플랫폼 버전에 대한 인지 전략과 로드맵을 파악하는 것이 중요합니다.

조절 가능한 자동화(Dialable Automation)

보안 운영과 자동화의 통합은 시간이 지남에 따라 증가하는 것이 일반적입니다. 대부분의 경우 플랫폼에 대한 신뢰를 얻어가는 과정으로서, 보안팀은 한 번에 하나의 유즈케이스에 대해서 자동화를 실시합니다. 자동화에 대한 신뢰 구축을 위해서는 자동화된 플레이북과 사용자의 선택적 상호 작용을 가능하게 도와줄 수 있는 일련의 기능을 SOAR 플랫폼이 지원해야만 합니다. 각 자산 단위(포인트 보안 도구 또는 기술)에서 혹은 각 액션 단위에서 워크플로우에 사용자가 개입할 수 있도록 해야만 합니다.

각 자산 단위에 대한 사용자의 개입을 위해서는, 해당 자산에 대해 액션이 실행될 때마다 관리자가 알림을 받을 수 있도록 해야 합니다. 각 액션 단위에 대한 사용자의 개입을 위해서는, 자동화된 플레이북의 어느 지점에서든 사용자의 즉각적인 액션을 추가할 수 있어야 합니다. 또한 사용자의 개입 지점에서 수신자가 액션 실행을 계속, 일시 중지, 중단할 수 있는 옵션을 제공해야 합니다. 이 같은 수준의 감독 활동이 가능해야 사용자는 프로그래밍된 자동화 액션에 확신을 얻을 수 있습니다.

보안(Secure)

보안 자동화 및 오케스트레이션 플랫폼의 가장 중요한 요소 중 하나는 플랫폼 자체의 보안입니다. SOAR 플랫폼은 인증 자격 증명 및 매우 민감한 정보를 보유하고 있기 때문에 시스템 하드닝, 민감 정보 암호화, 강력한 역할 기반 접근 제어 기능을 제공합니다.

따라서 SOAR 플랫폼은 다음과 같은 주요 보안 요소를 필요로 합니다.

- 암호화된 보안 자격 증명
- 자격 증명이 메모리에 저장되지 않게 보증
- 인증 관리 시스템 지원
- 다중 인증 지원

확장성(Scalable)

또한 SOAR 플랫폼이 어떻게 수직 및 수평 확장성을 구현하는지 이해하는 것이 중요합니다. 첫 도입 이후 시간이 지남에 따라 유즈케이스가 증가할 것이고, 플랫폼에 더 많은 처리 부하가 발생할 것입니다. 따라서 하드웨어 리소스(CPU 및 RAM 등)를 늘리는 수직적 규모 확장성과 배포를 지원하는 서버 인스턴스의 수를 늘리는 수평적 확장성이 가능하도록 플랫폼을 설계해야 합니다.

개방성과 유연성(Open and Extensible)

보안은 항상 진화하고 있으며, 다양한 포인트 제품들을 보면 빠른 변화를 실감할 수 있습니다. 따라서 SOAR 플랫폼은 개방성과 추가 가능성을 고려해 설계되어야 합니다. 이를 통해 새로운 보안 시나리오, 새로운 제품, 새로운 액션 및 새로운 플레이북의 통합을 쉽게 지원해야 합니다.

개방형 통합 프레임워크(Open Integration Framework)

개방형 통합 프레임워크의 진정한 효과를 누리기 위해서는, 보안 기술이 자동화된 운영에 부정적인 영향을 미치지 않으면서 동시에 플랫폼에 기술의 추가 및 삭제가 자유로워야 합니다.

구체적으로 사용자는 SOAR 플랫폼 벤더에 의존하지 않고, 추가 통합에 대한 지원 사항 개발의 통제 권한이 있어야 합니다. 달리 말해, 사용자가 원하는 경우 자체 통합을 구현할 수 있어야만 합니다. 자체 개발 애플리케이션, 벤더의 커스텀 또는 얼리 액세스 API, 또는 기존 자동화 플랫폼의 기능 확장 등이 개방형 통합 프레임워크가 적용될 수 있는 좋은 예시입니다.

이러한 개방형 프레임워크는 공통 표준 및 프로그래밍 모델을 따라야 합니다. 또한 가용한 문서와 예시가 풍부해야 합니다.

인터페이스 제한 유무(No Interface Restrictions)

일부 기술은 REST API, SSH, syslog, 커스텀 API 또는 기타 프로토콜이나 도구를 사용하여 인터페이스를 보여줍니다. 하지만 확장 가능한 통합 프레임워크는 인터페이스 유형에 대한 제한이 있어서는 안 됩니다. 자동화 플랫폼에서 포인트 제품 또는 애플리케이션으로 연결이 필요한 경우, 인터페이스 방식이 앱 통합에 영향을 미치면 않으며, 모든 인터페이스 도구를 허용해야만 합니다.

모바일(Mobile)

SOAR 플랫폼은 대응 시간을 가속하도록 설계되었습니다. 즉, 체류 시간(MDT)과 평균 해결 시간(MTTR)을 줄이는 것이 목표입니다. 이러한 신속한 대응을 위해서는 보안 분석가가 사용자의 개입이 필요한 케이스 또는 보안 조치 지점에 접근할 수 있어야 합니다. 그러나 분석가가 노트북을 켜고 즉시 응대할 준비가 항상 되어 있는 것은 아닙니다. 이러한 이유에서 SOAR 플랫폼에서 분석가가 자신의 모바일 기기로 편리하게 플랫폼에 대한 접근, 상호 작용, 제어를 할 수 있도록 해주어야 합니다. 모바일 디바이스를 사용하면 분석가는 이동 중에 플레이북을 실행하고, 노트북 없이 보안 아티팩트 검토 및 이벤트를 분류하는 등 모바일 기기에서 응대가 가능합니다. 또한 분석가가 어디에 있든지 항상 연락이 가능합니다.

사용 용이성(Ease-of-Use)

전사적 소프트웨어가 단순한 경우는 결코 없지만 SOAR 플랫폼을 배포하고 사용할 때 발생할 수 있는 문제를 경감하는 것은 가능합니다.

설치 및 설정(Installation and Setup)

대부분의 조직에서 이미 다른 인프라와 함께 가상화를 활용하고 있으므로 가상 어플라이언스 폼 팩터를 사용하면 배포가 손쉬워집니다.

신규 사용(Onboarding)

SOAR 플랫폼은 온보딩 프로세스를 사용하여 사용자가 시스템 설정을 구성하고, 데이터 소스(연동 장비 또는 로그)를 연결하고, 최초의 플레이북 몇 개를 활성화하게 하는데 도움을 제공하기 때문에, 사용자가 SOAR플랫폼에 적응하는데 걸리는 시간을 줄여 줍니다.

자동화 시간 단축(Accelerate the Time-to-Automate)

SOAR 플랫폼은 사용자가 자동화를 빠르게 시작할 수 있도록 도움을 주어야 합니다. 이 목표를 달성하기 위해서는 즉시 사용할 수 있는 강력한 자동화 플레이북 세트를 제공해야 합니다. 또한 사용자가 자동화된 플레이북을 신속하게 작성, 테스트 및 배포할 수 있도록 하는 것 역시 자동화를 위해서 중요합니다.

비주얼 에디터를 사용하면 프로그래머가 아니더라도 손 쉽게 플레이북을 생성하고 수정할 수 있습니다. 따라서 SOAR 플랫폼은 자동화 플레이북을 시각적으로 생성하거나 수정하는 툴을 제공하여 비개발자의 업무에 도움을 주어야 합니다. 추가적으로, 플레이북을 시각적으로 구성하면 코딩 오류를 방지하여 개발 시간을 단축하고 품질을 향상하며, 플레이북 전체의 일관성을 유도하는 코딩 표준을 제공할 수 있습니다.

비즈니스 고려사항

한 업체의 핵심 기술이 아무리 뛰어나더라도 전통적으로 구매자의 의사 결정 과정에 크게 영향을 끼치는 제품 외적인 요소들이 존재하게 마련입니다. 그 중 한 가지가 바로 제품과 서비스를 제공하는 회사가 어떠한 회사인가 하는 점입니다. 또 다른 요소는 전체 제품군을 구성하는 핵심 기술을 사용하는 구매자의 최종 경험을 한 층 강화할 수 있도록 어떠한 서비스를 제공하는가 여부입니다.

업체의 특질(Company Attributes)

구매 및 조달 결정을 내릴 때 선택한 업체의 프로필, 품질, 미래 잠재력을 고려하는 것이 중요합니다. 그러나 새로운 시장 영역에 새로운 솔루션을 제시하는 많은 신생 업체가 존재하고 결국 문을 닫는 경우가 많은 것이 현실입니다. 따라서 자신의 약속을 실현할 수 있는 충분한 능력을 보유한 업체를 선택해야 합니다.

업체의 연혁(Company History)

보안 솔루션 개발에 깊이 있는 역사를 가지고 있는 업체를 선택할 필요가 있습니다. 보안 오케스트레이션, 자동화 및 대응은 비교적 새로운 시장 영역이지만 그 기원은 수십 년으로 거슬러 올라갈 수 있습니다. 결국 해당 업체가 어떻게 형성되었고 SOAR 영역에 뛰어들기로 결정했는지를 이해하는 것이 중요합니다.

실행 능력(Ability to Execute)

경험 많은 전문가들로 구성된 노련한 보안 팀이 뒷받침하는 업체를 찾아야 합니다. 한 업체의 미래 실행 능력은 팀 구성원의 과거 실적과 직접적 상관관계를 가질 확률이 매우 높습니다.

고객 기반(Customer Base)

업체가 지닌 고객의 수준과 프로필은 해당 업체의 상황을 잘 보여주는 요소입니다. 일을 정밀하게 하는 기업 고객은 구매하기 전에 여러 영역에서 후보 업체에 대해 엄격한 실사를 수행합니다.

수상 및 성과(Awards and Recognition)

업체의 수상 실적과 그 외에 어떤 성과를 인정받았는지 확인하십시오. 업체가 주장하는 우수성과 과거의 수상 실적과 성과가 일치하는지 여부를 살펴볼 필요가 있습니다. 또한 회사의 수준이 다양한 것과 마찬가지로 모든 상이 같은 무게를 지니는 것은 아닙니다.

부가 서비스(Ancillary Services)

업체가 자사의 기술에 대해 제공하는 부가 서비스는 구매자의 배포 경험과, 궁극적으로는 프로젝트의 성공에 큰 영향을 미칠 수 있습니다.

전문 서비스(Professional Services)

보안 운영의 성숙도 수준은 조직마다 크게 다를 수 있습니다. 따라서 해당 업체가 전문 서비스를 제공하여 성공적 배포의 가능성을 높일 수 있는지 확인하는 것이 중요합니다. 또한 프로세스 구축(내부 인원이 부족한 경우에)을 돕고 수동 워크플로우를 자동화 플레이북으로 변환하는 데 도움이 되는 전문가를 제공할 수 있는지 여부의 확인도 중요합니다.

판매 후 지원(Post-Sales Support)

많은 신생 업체가 우수한 기술과 판매 전 지원을 제공하지만 판매 후 지원에 있어서는 그렇지 않습니다. 지원 옵션의 범위를 조사하고 고객사가 필요한 지원 유형을 제공하는지 여부를 고려하십시오.

3. 결론

본 가이드에는 SOAR 플랫폼을 평가할 때 고려해야 할 많은 기준이 포함되어 있습니다. 핵심 기능, 플랫폼 속성, 비즈니스 고려 사항에 이르기까지 평가 기준을 완전히 준비하고, 그 다음에 평가 프로세스를 시작하고 플랫폼을 선택하는 것이 중요합니다.

Splunk SOAR 보안 자동화 및 오케스트레이션 플랫폼에 대해 자세히 알아보려면 Splunk SOAR Community Edition을 무료 다운로드 하거나 영업팀에 자세한 정보를 문의하십시오.

Splunk, Splunk>, Data-to-Everything, D2E and Turn Data Into Doing are trademarks and registered trademarks of Splunk Inc. in the United States and other countries. All other brand names, product names or trademarks belong to their respective owners. © 2021 Splunk Inc. All rights reserved.

21-13321-SPLK-SOAR Buyers Guide-109-WP

평가 체크리스트

다양한 SOAR(보안 오케스트레이션 및 자동화, 대응)플랫폼을 평가할 때 아래의 간편한 체크리스트를 사용하십시오.

플랫폼 이름: _____

핵심 기능

☐ 오케스트레이터

- ☐ 데이터 수집
- ☐ 의사결정
- ☐ 태스크 실행
- ☐ 사용자의 감독
- ☐ 데이터 관리
- ☐ 무중단 운영

☐ 자동화 엔진

- ☐ 성능 확장성(Scalability)
- ☐ 유연성(Extensibility)

☐ 경보 관리

- ☐ 경보 상세 정보
- ☐ 액션 발생
- ☐ 액션 결과
- ☐ 활동 로그
- ☐ 경보 상태, 심각도 및 민감도
- ☐ 경보 협업

☐ 케이스 관리

- ☐ 케이스 데이터 구성
- ☐ 케이스에 데이터 추가
- ☐ 경보에 케이스 연결
- ☐ 기존 프로세스에 매핑
- ☐ 활동 감사

☐ 플레이북 관리

- ☐ 플레이북 구성
- ☐ 커스텀 함수
- ☐ 플레이북 대량 편집
- ☐ 형상 관리 및 배포

☐ 자동화 에디터

- ☐ 사용자 화면(UI) 요소
- ☐ 코드의 블록 기반 표현
- ☐ 의사결정 과정에 사용자 개입
- ☐ 액션 결과의 정보 교환
- ☐ 플레이북 소스 코드에 대한 접근
- ☐ 시각적 및 비시각적 플레이북 동시 구축
- ☐ 내장형 테스트와 디버깅 및 런타임 로깅
- ☐ 안전 모드

☐ 앱 프레임워크

- ☐ 개방형 생태계

☐ 메트릭스 및 보고

- ☐ 유연한 대시보드
- ☐ 성과 보고
- ☐ 보안 효율성 보고
- ☐ 앱 통합 및 플레이북 성능
- ☐ 인적 작업량

플랫폼 속성

☐ 배포 옵션

- ☐ 온프레미스
- ☐ 클라우드
- ☐ 하이브리드

☐ 커뮤니티 기반

- ☐ 크고 활동적인 커뮤니티

☐ 협업

- ☐ 커뮤니티 전반의 협업
- ☐ 플랫폼 전반의 협업

☐ 인지

- ☐ 조절 가능한 자동화

☐ 보안

☐ 확장성

☐ 개방성과 추가 가능성

☐ 개방형 통합 프레임워크

☐ 인터페이스 제한 유무

☐ 사용 용이성

- ☐ 설치 및 설정
- ☐ 신규 사용
- ☐ 자동화 시간 단축

비즈니스 고려사항

☐ 업체의 특질

☐ 업체의 연혁

☐ 실행 능력

☐ 고객 기반

☐ 수상 및 성과

☐ 부가 서비스

☐ 전문 서비스

☐ 판매 후 지원