Splunk4Ninjas - Machine Learning for Security

2 days Security Camp

2024.11

강채희 / Solutions Engineer

ckang@splunk.com

Forward-looking statements

This presentation may contain forward-looking statements regarding future events, plans or the expected financial performance of our company, including our expectations regarding our products, technology, strategy, customers, markets, acquisitions and investments. These statements reflect management's current expectations, estimates and assumptions based on the information currently available to us. These forward-looking statements are not guarantees of future performance and involve significant risks, uncertainties and other factors that may cause our actual results, performance or achievements to be materially different from results, performance or achievements expressed or implied by the forward-looking statements contained in this presentation.

For additional information about factors that could cause actual results to differ materially from those described in the forward-looking statements made in this presentation, please refer to our periodic reports and other filings with the SEC, including the risk factors identified in our most recent quarterly reports on Form 10-Q and annual reports on Form 10-K, copies of which may be obtained by visiting the Splunk Investor Relations website at www.investors.splunk.com or the SEC's website at www.sec.gov. The forward-looking statements made in this presentation are made as of the time and date of this presentation. If reviewed after the initial presentation, even if made available by us, on our website or otherwise, it may not contain current or accurate information. We disclaim any obligation to update or revise any forward-looking statement based on new information, future events or otherwise, except as required by applicable law.

In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not be incorporated into any contract or other commitment. We undertake no obligation either to develop the features or functionalities described, in beta or in preview (used interchangeably), or to include any such feature or functionality in a future release.

splunk>

Workshop Agenda

- What is Machine Learning이란 무엇일까요?
- 보안 분야에서 Machine Learning이 왜 필요할까요?
- Machine learning for security with Splunk
- 실습 진행
- ML을 활용한 어플리케이션 만들기

Enroll in Today's Workshop

Tasks

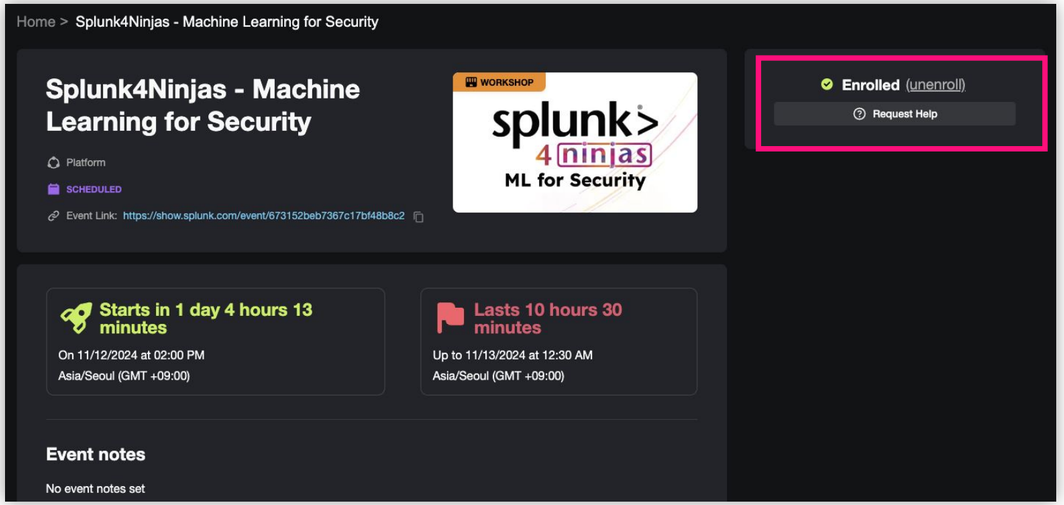
1. Splunk 계정이 없으시다면 Splunk 계정을 만들어야 합니다.
   https://splk.it/SignUp
2. 아래의 링크로 접속해 workshop 이벤트를 등록하세요.
   https://show.splunk.com/event/673152beb7367c17bf48b8c2
3. 아래의 lab guide를 다운 받으세요
   https://drive.google.com/drive/folders/1LggmqjeinBi95cw0U0AbEFC11rR3ssgB?usp=sharing

https://github.com/chgchookang/S4N_ML4Security

오늘 워크샵의 모든 단계에 대한 안내 문서입니다.

Goal

Enroll in today's event

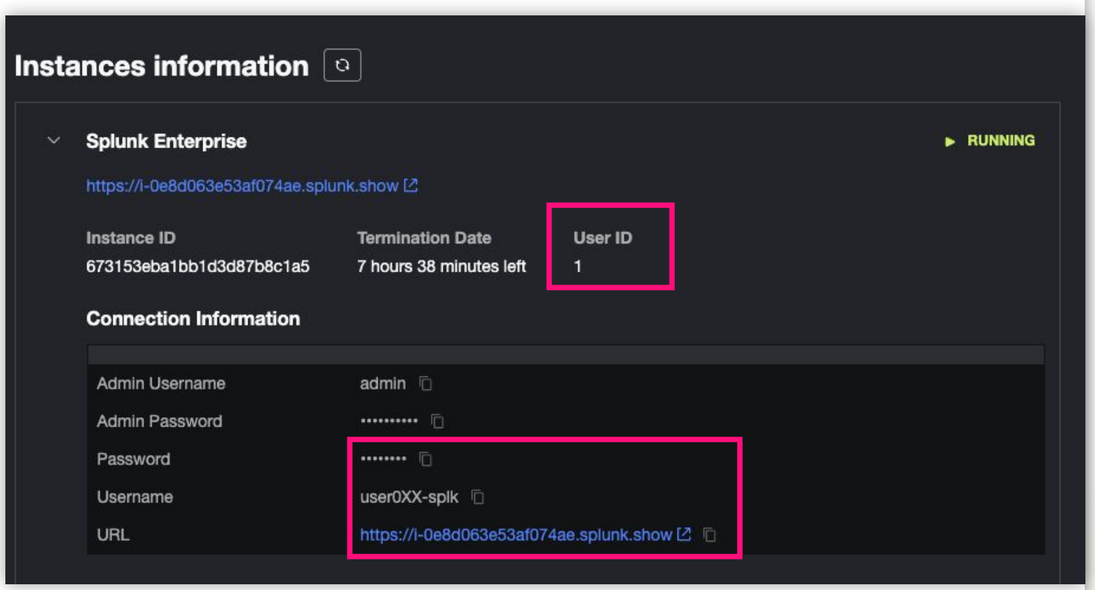


Enroll in Today's Workshop

Tasks

4. 하단의 Instances Information에서 ID와 PWD 확인
   - ID의 X 값은 User ID에 있는 값으로 대체
   - Ex) User ID가 2라면 id는 user002-splk

Goal



DOWNLOAD MATERIALS 5 MINS

Obtain the Materials for Today's Workshop

Tasks

1. Get your instan
   <link to Gsheet
2. Download a cop
   https://splk.it/M

Goal

701533c62d97ab6.splunk.show
527c2526e8e3712.splunk.show
02422ed1f20c670.splunk.show
f3ac2950177bc25.splunk.show
475a6ebca7d779e.splunk.show

name against one
ances in the list

**** Optional slide for running a 'normal' workshop ****
(i.e. not an 'event') in Splunk Show

Presenter instructions:

1. **Create a 'normal' workshop in Splunk Show** selecting the required number of instances you require for your workshop (see the Splunk Show User Guide for Splunkers or for Partners)
2. Once the instances are all running, **export a CSV of the instances**
3. **Share the list of instances via Google Sheets** (example Gsheet here) or some other method that will allow attendees to obtain their own instance from the list.
4. **Copy the URL for the Gsheet into step 1 of this slide**
5. **Skip the previous slide** and **unskip this slide**!
6. During the workshop **ask users to put their names against a free instance in the Gsheet** as a way of tracking who is allocated to which instance
7. **Delete or move this text box off screen** before presenting!

Machine Learning이란 무엇일까요?

splunk>

AI와 Machine Learning이란 무엇일까요

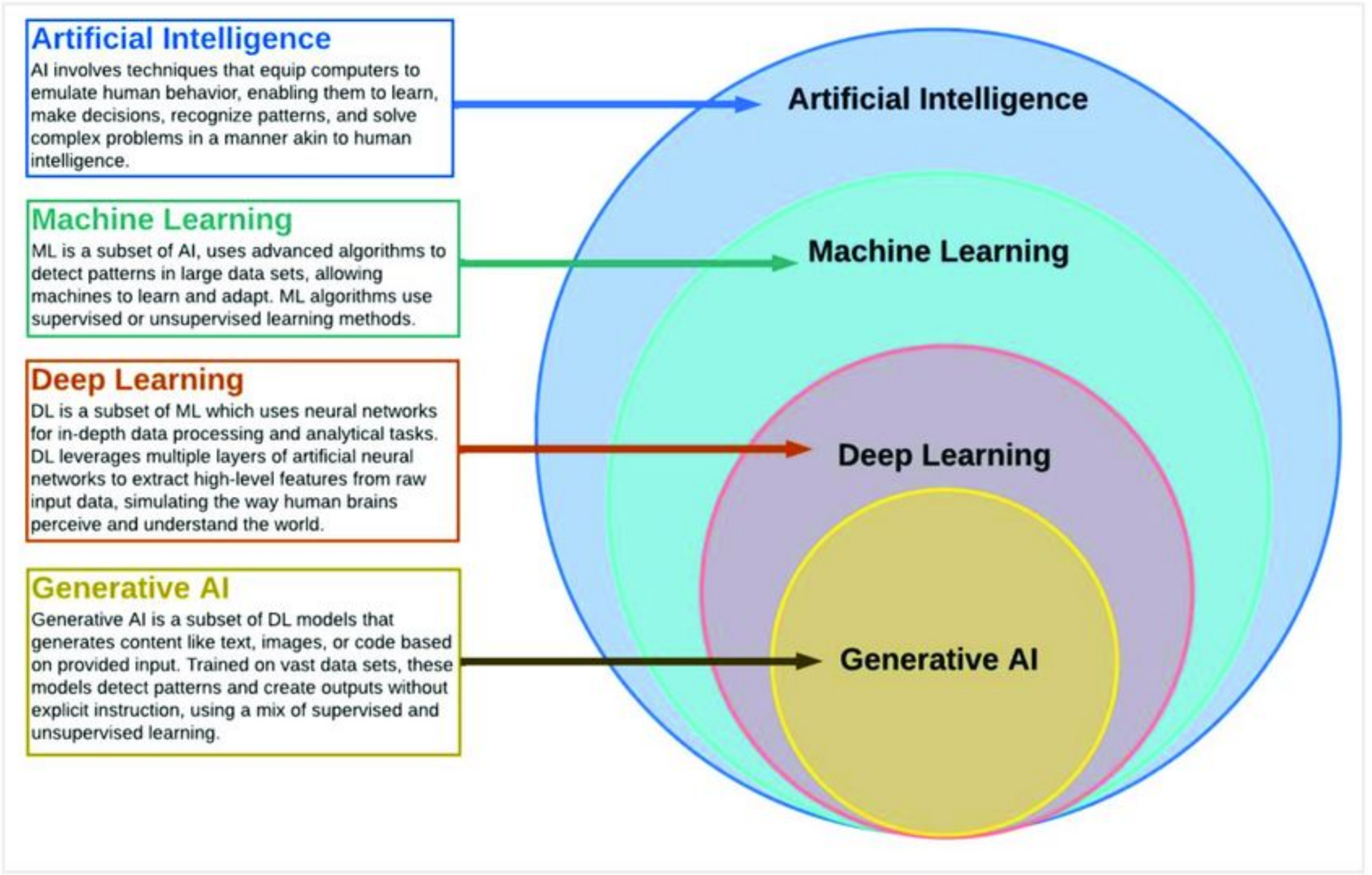
Definitions

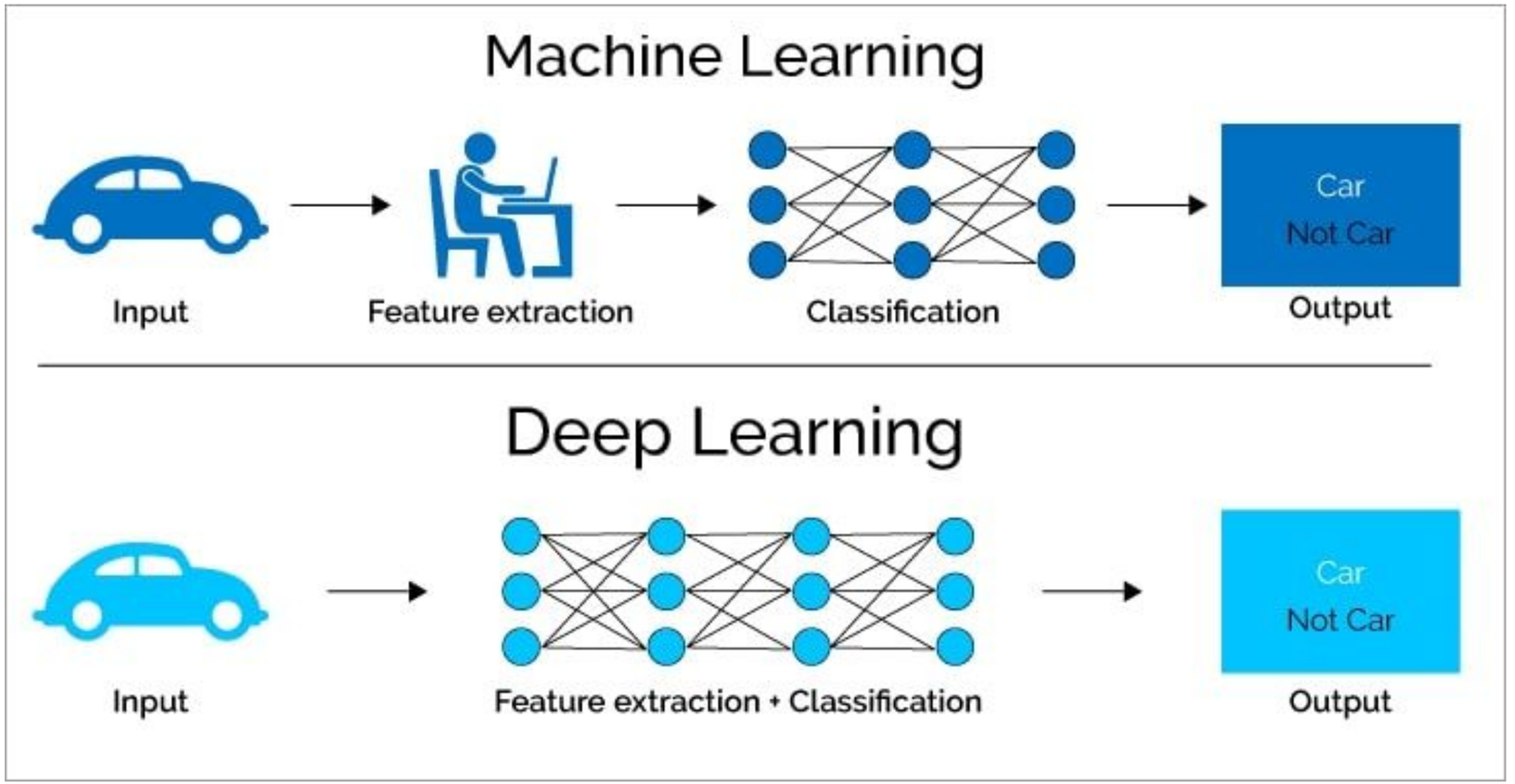
Artificial Intelligence (AI) - 학습 및 문제 해결과 같은 **인간의 인지 기능을 모방**하는 컴퓨터 시스템의 기능

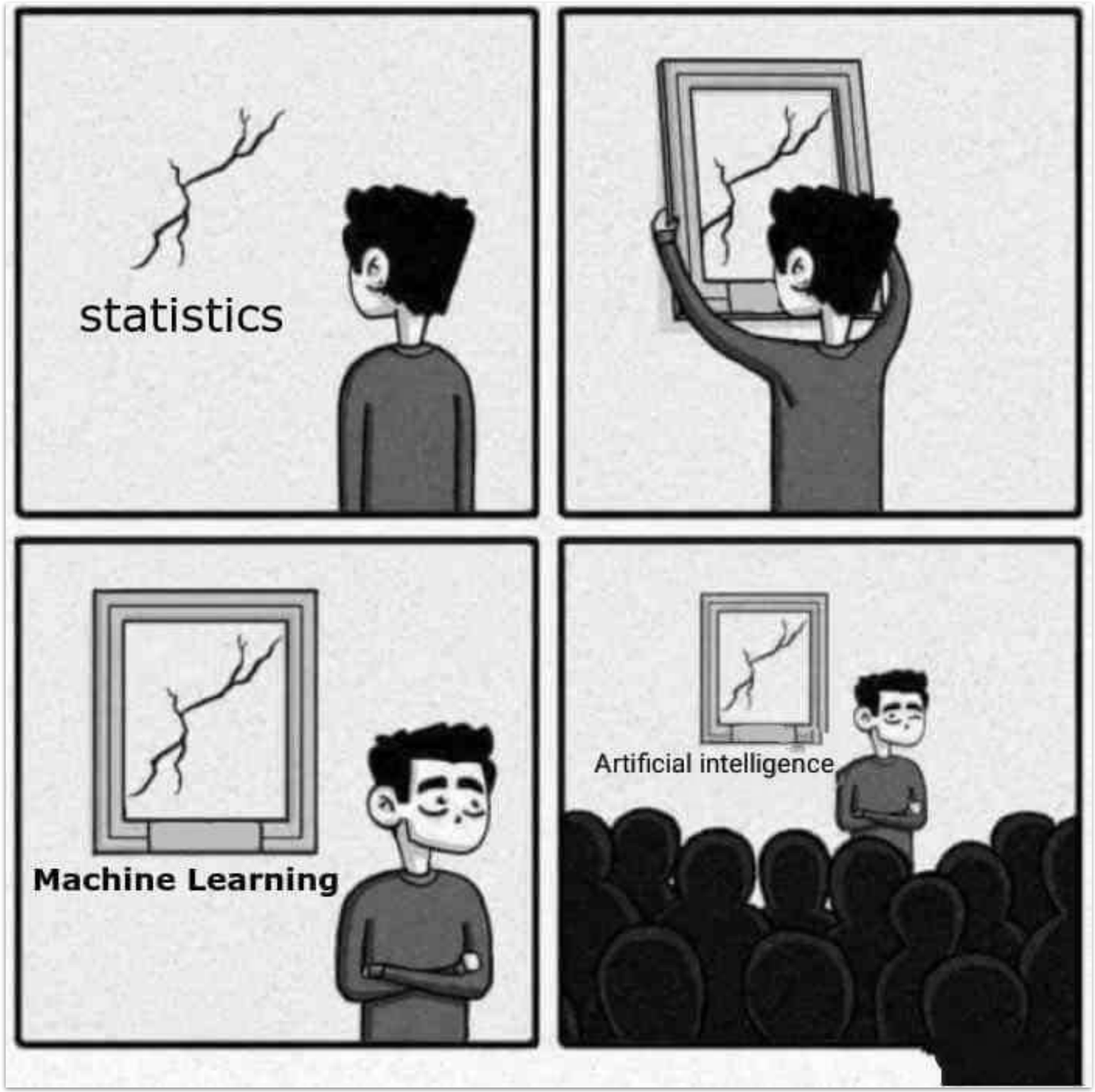
Machine Learning (ML) - 데이터에서 패턴을 학습하고, 이를 바탕으로 예측이나 분류를 수행하는 알고리즘과 기술 - 지도 학습, 비지도 학습, 강화 학습 -

Deep Learning - 머신 러닝의 하위 분야로 인공 신경망을 사용하는 알고리즘 이미지, 음성과 같은 비정형 데이터를 처리하는데 뛰어남 *예: 자율 주행 차량이 정지 신호를 인식*

Generative AI - 알고리즘과 기술을 사용하여 모델에 의해 생성되기 전에는 세상에 존재하지 않았던 새로운 데이터를 생성하는 **AI의 하위 집합** *예시: OpenAI의 ChatGPT*







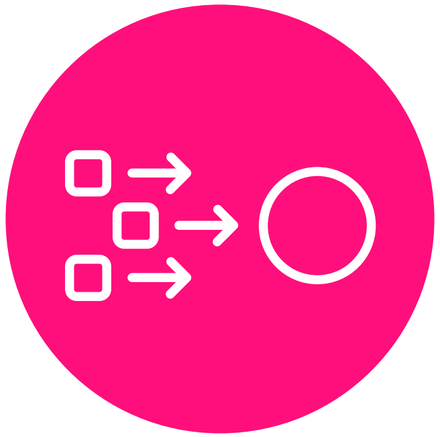
© Thomas Wiecki, Ph.D., ODSC Europe 2018

Machine learning (ML)은
경험과 과거 행동을 통해 자동으로 개선되는
컴퓨터 알고리즘을 연구하는 학문입니다.

- Wikipedia

Machine Learning

패턴을 추출하여 자체적으로 지식을 습득하는 시스템의 기능



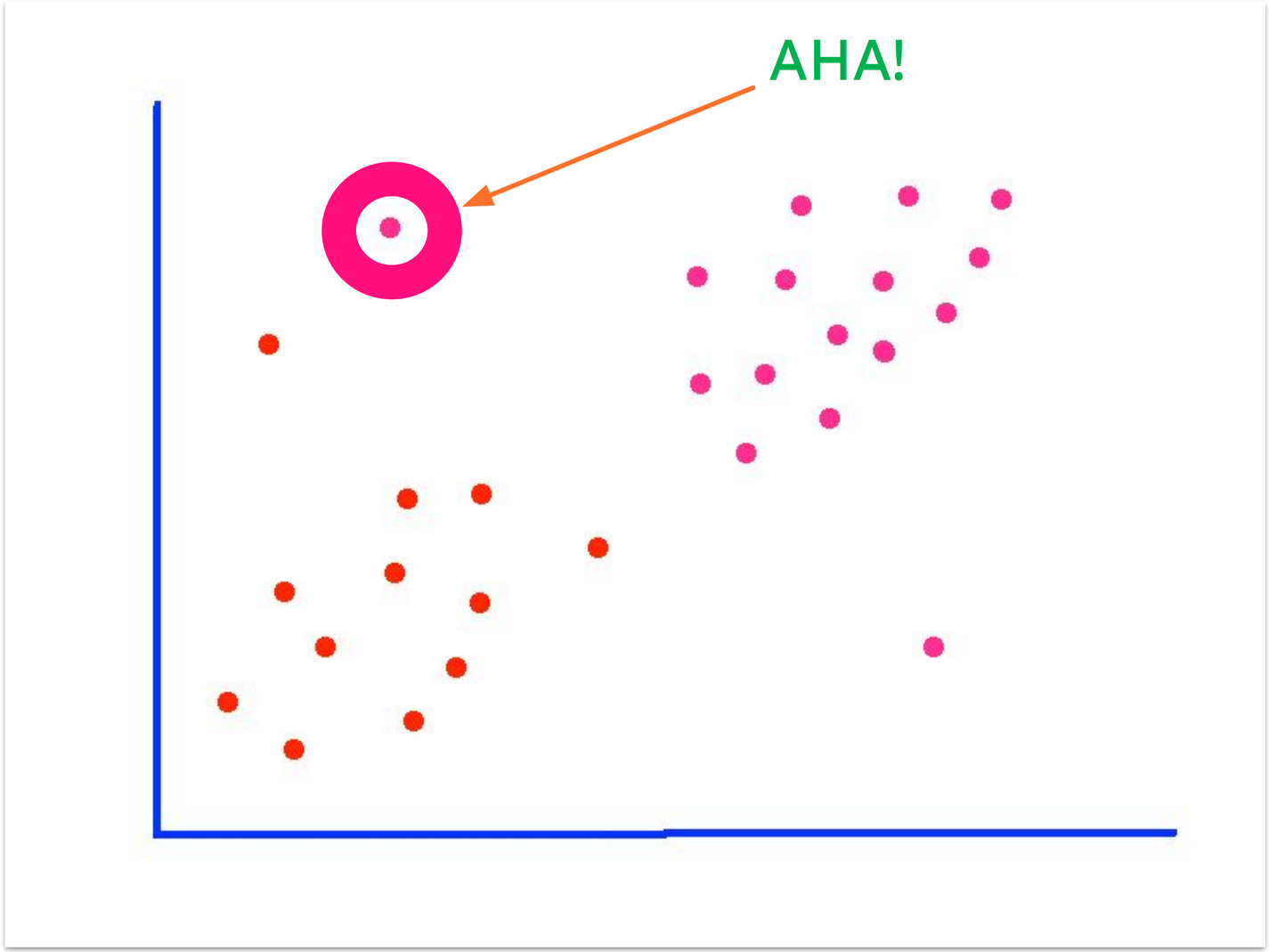
복잡한 통계 또는 확률 모델을 사용하여 정보의 패턴을 식별합니다.

패턴을 카탈로그(리스트)화하고 경우에 따라 새로운 데이터가 수신될 때 패턴을 반복합니다.

학습된 패턴의 정보를 사용하여 새로운 데이터를 이해하고 해석하거나 예측합니다.

Correlation

This is NOT ML



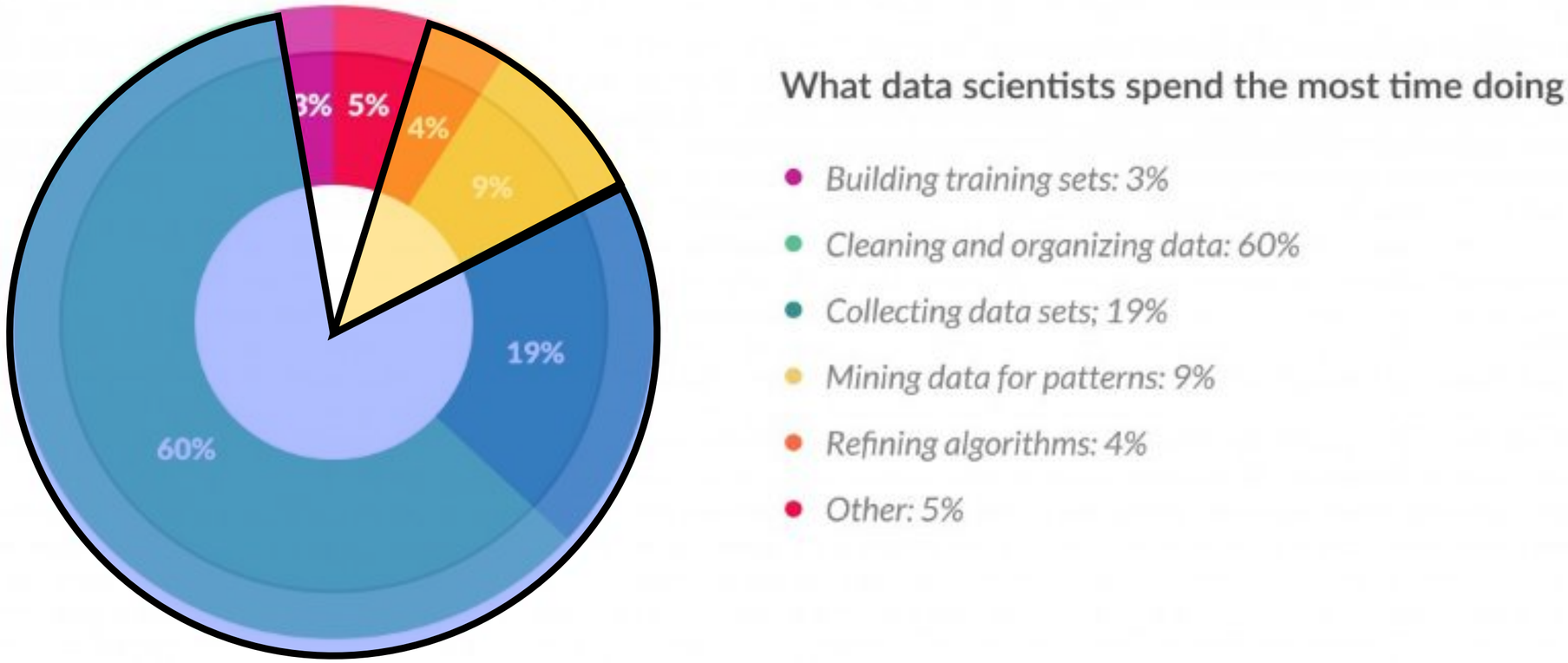
Machine Learning

왜 필요할까요?

- 평균 및 표준 편차를 이용한 예측과 같은 기본적인 통계 모델링 기법만으로는 이러한 질문에 정확하게 답할 수 있을 만큼 세분화되어 있지 않습니다.
- 머신 러닝(ML)은 복잡한 통계 기법을 사용하여 이러한 유형의 질문에 대해 더 나은 예측을 제공합니다.
- 예를 들어, 많은 조직에서 선제적 또는 타겟팅된 의사 결정을 내리는 데 데이터를 사용하는 것이 점점 더 중요해지고 있습니다:
  - 네트워크에서 봇넷의 존재 감지
  - 침해된 사용자 계정 탐지
  - 악성 계정 또는 비정상적인 활동을 탐지하여 뱅킹 사기 방지
  - 소셜 미디어에서 보다 타겟팅된 광고 캠페인을 위한 고객 분류
  - 공급망 관리를 지원하기 위한 제품 수요 예측

Data Scientists 해야할 일

데이터 준비 Data Scientists 업무의 약 80%를 차지합니다.



“Cleaning Big Data: Most Time-Consuming, Least Enjoyable Data Science Task, Survey Says”, Forbes Mar 23, 2016

ML Starter

해야할 것과 하지 말아야할 것

What Machine Learning에 대한 오해

- 한 가지 크기로 모든 데이터 문제(일명 매직 버튼)에 대한 모든 답변을 확인할 수 있습니다
- 인공 지능...
- 예측 명령
- 상관 규칙

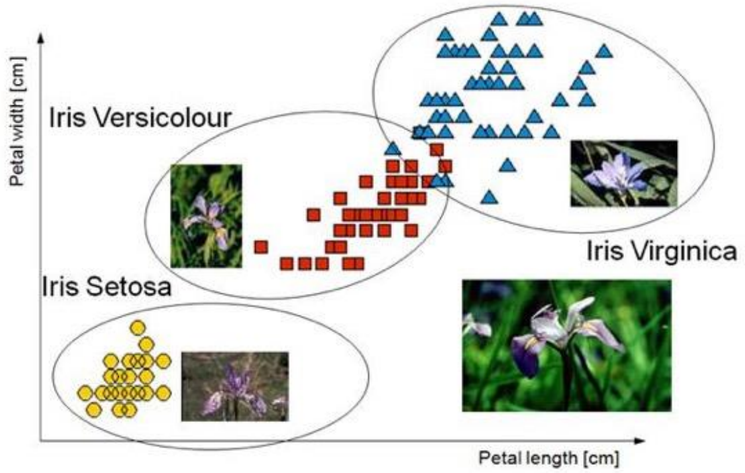
What Machine Learning에 대한 진실

- 향후 탐지를 위해 데이터에서 패턴을 발견하고 기준을 설정하는 좋은 방법 제시
- 정의가 명확하게 되어있고 경계가 뚜렷하며 사용해야 하는 데이터를 명확하게 이해하는 경우

Machine Learning의 종류

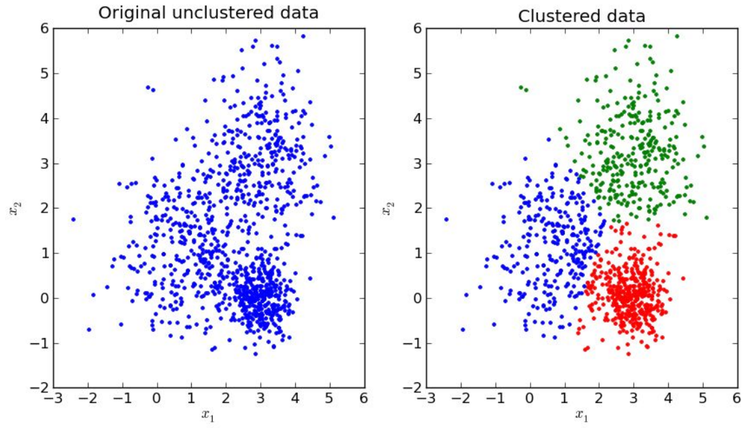
지도 학습 (labeled data)

- Regression(회귀)
- Classification(분류)



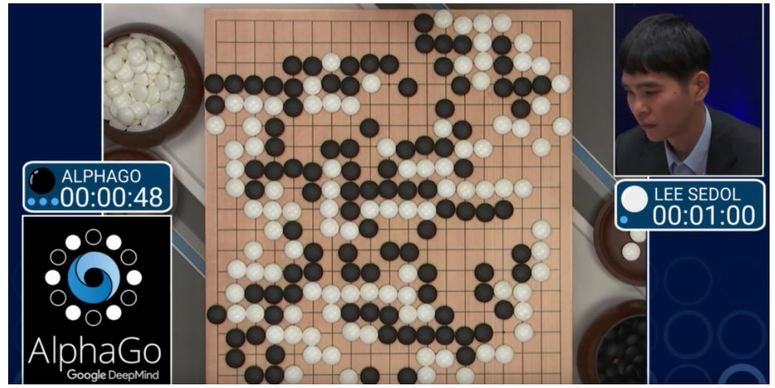
비지도 학습 (unlabeled data)

- Clustering(군집화)
- Anomaly Detection(이상 탐지)



강화 학습 (with reinforcement or feedback)

- Human in the Loop
- Autonomous Systems



왜 보안 분야에서 ML이 필요할까요?

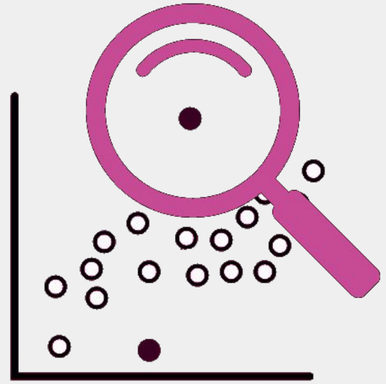
splunk>

SOCs 의 집중 분야와 변화된 역할

LEGACY	REQUIRED
상황 인식	분석 및 의사 결정
운영/모니터링 센터	**Center** 조절 및 지휘
인간이 중심인	인간 — **Machine Learning**
Human Speed Operations	**Machine-Speed Cycle Times**

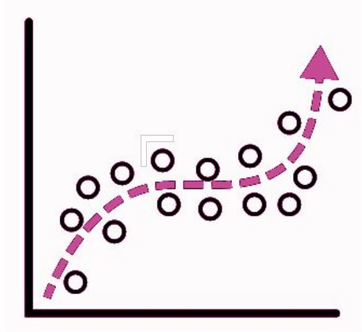
Splunkers 가 데이터로 도출하고 싶은 결과는?

Anomaly detection - 이상 탐지



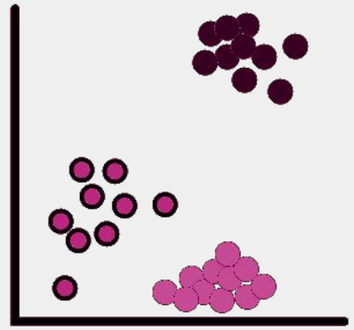
- 과거 행동으로부터의 이탈
- 이탈 (Multivariate AD 또는 Cohesive AD로도 알려짐)
- 기능에서의 비정상적인 변화

Predictive Analytics - 예측 분석



- 서비스 상태 점수/이탈 예측
- 이벤트 예측
- 트렌드 예측
- 영향을 끼치는 엔티티 탐지
- 고장 조기 경고

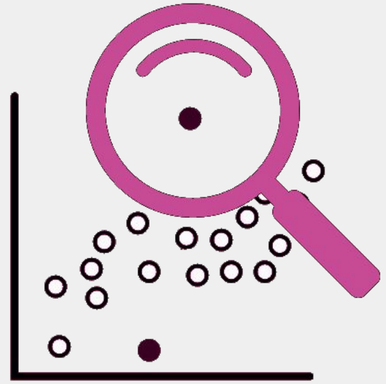
Clustering - 군집화



- 그룹 식별
- 이벤트 상관관계
- 경보 감소
- 행동 분석

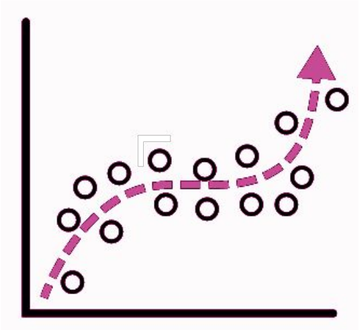
Splunk 고객들이 그들의 데이터로 도출하고 싶은 것?

Anomaly detection - 이상 탐지



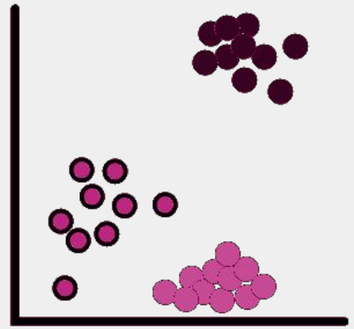
- 원하지 않거나 예상치 못한 행동을 탐지할 때 유용합니다.
- 장기간에 걸쳐 천천히 진행되는 저속 공격을 탐지할 때 유용합니다.

Predictive Analytics - 예측 분석



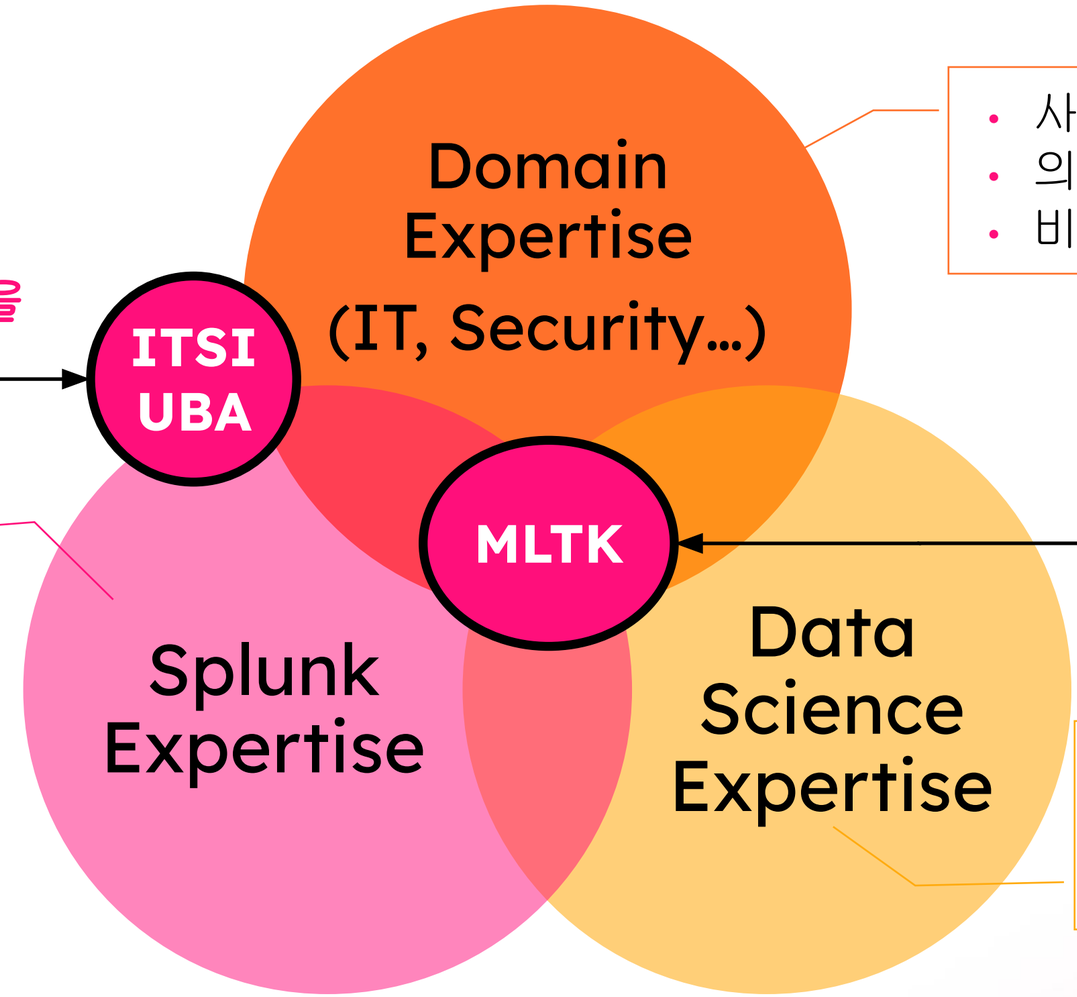
- 이전 반응을 바탕으로 경고의 결과(FP/TP)를 예측합니다.
- 이전 반응을 바탕으로 대응 권장사항을 생성합니다.

Clustering - 군집화



- 관찰된 이벤트를 기반으로 동적 동료 그룹을 생성하는 데 사용됩니다.
- 유사한 경고를 그룹화하는 데 사용됩니다.

Skill Areas for Machine Learning @ Splunk



- 사용 사례 식별
- 의사 결정 주도
- 비즈니스 영향 이해

프리미엄 솔루션은 기본 제공되는 머신 러닝 기능을 지원함

- Searching
- Reporting
- Alerting
- Workflow

예시와 가이드를 통해 기능을 제공하고 사용을 간소화함

- Statistics/math background
- Algorithm selection
- Model building

Custom ML with the Splunk Platform

Ecosystem

Splunk의 앱 에코시스템에는 데이터를 가져오고, 구조를 적용하며, 데이터를 시각화하는 데 필요한 수천 개의 무료 Add-on이 포함되어 있어 더 빠르게 가치를 실현할 수 있습니다.

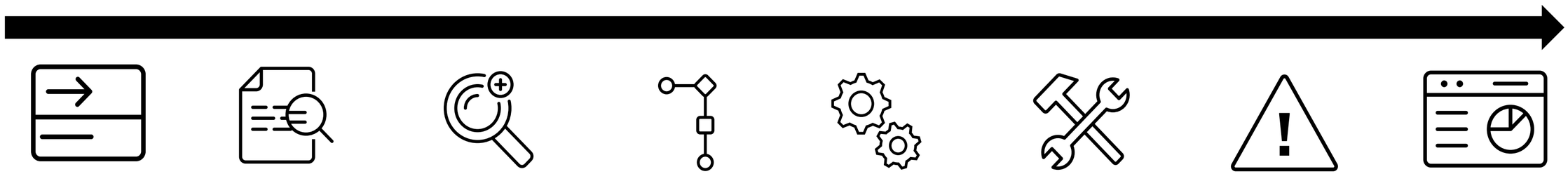
MLTK

머신 러닝 툴킷은 새로운 SPL 명령, 사용자 지정 시각화, 어시스턴트 및 다양한 ML 개념을 탐색할 수 있는 예제를 제공합니다.

Splunk

Splunk Enterprise는 인덱싱, 검색, 분석, 경고 및 머신 데이터 시각화를 위한 중요한 플랫폼입니다.

Data Science 파이프라인



Data 수집	정리 & 처리	Search & Explore	전처리 및 특징 선택	Algorithm 선택	모델 구축, 테스트, 개선	모니터링 및 경고	시각화 & 공유
Ecosystem	Ecosystem	Splunk	MLTK	MLTK	MLTK	MLTK	Ecosystem
Splunk	Splunk		Splunk	Splunk	Splunk	Splunk	Splunk

splunk> Platform for Operational Intelligence

The Security Data Journey

Advanced Detection - 고급 탐지
머신 러닝을 포함한 정교한 탐지 메커니즘 적용

Stage 6

Automation and Orchestration - 자동화 및 오케스트레이션
일관되고 반복 가능한 보안 운영 능력 수립

Stage 5

Enrichment - 보강
보안 데이터를 인텔리전스 소스와 결합하여 이벤트의 맥락과 영향을 더 잘 이해

Stage 4

Expansion - 확장
엔드포인트 활동 및 네트워크 메타데이터와 같은 고품질 데이터 소스를 추가하여 고급 공격 탐지를 강화

Stage 3

Normalization - 정규화
표준 보안 분류 체계를 적용하고 자산 및 신원 데이터를 추가

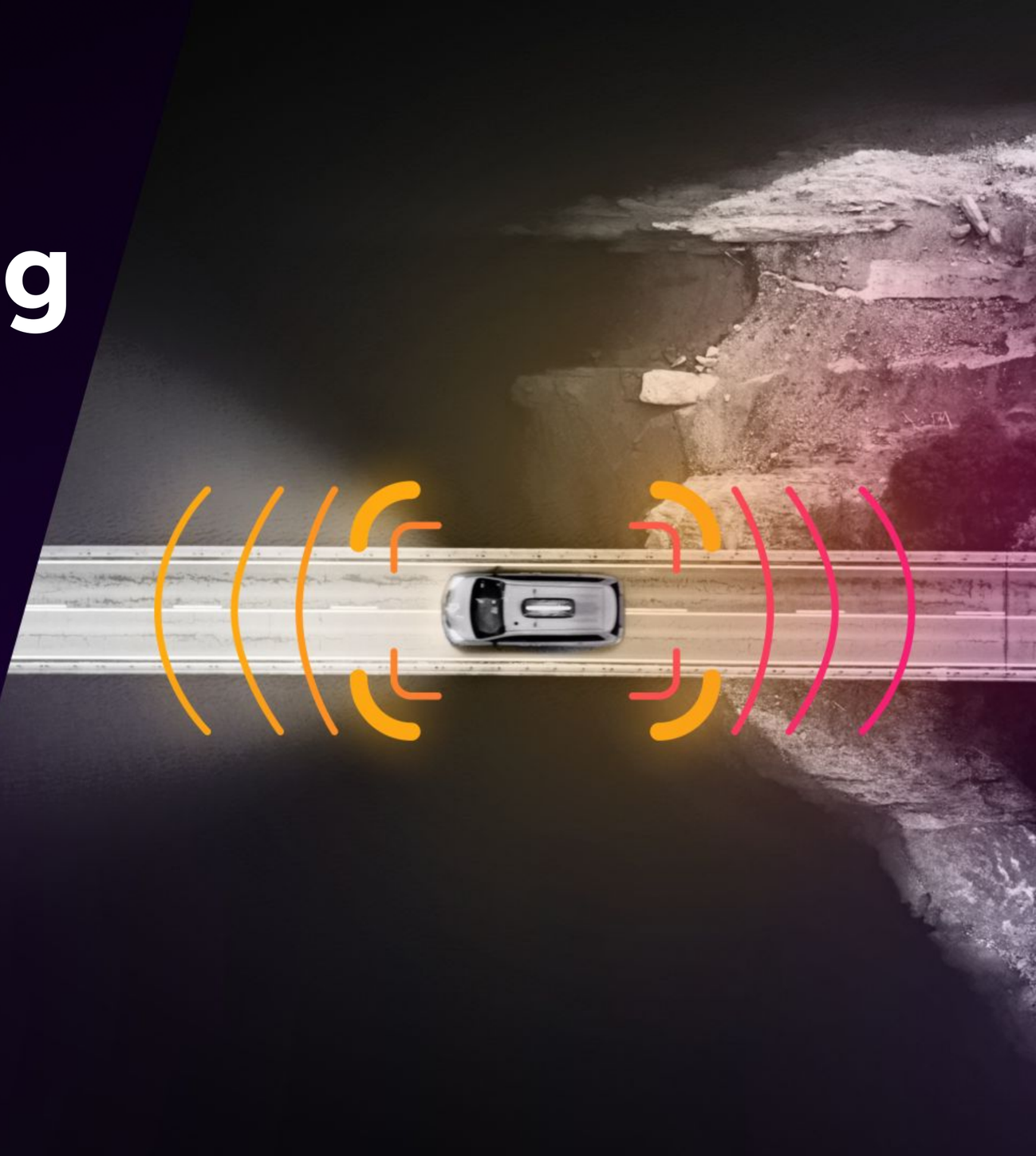
Stage 2

Collection - 수집
기본 보안 로그 및 기타 기계 데이터를 환경에서 수집

Stage 1

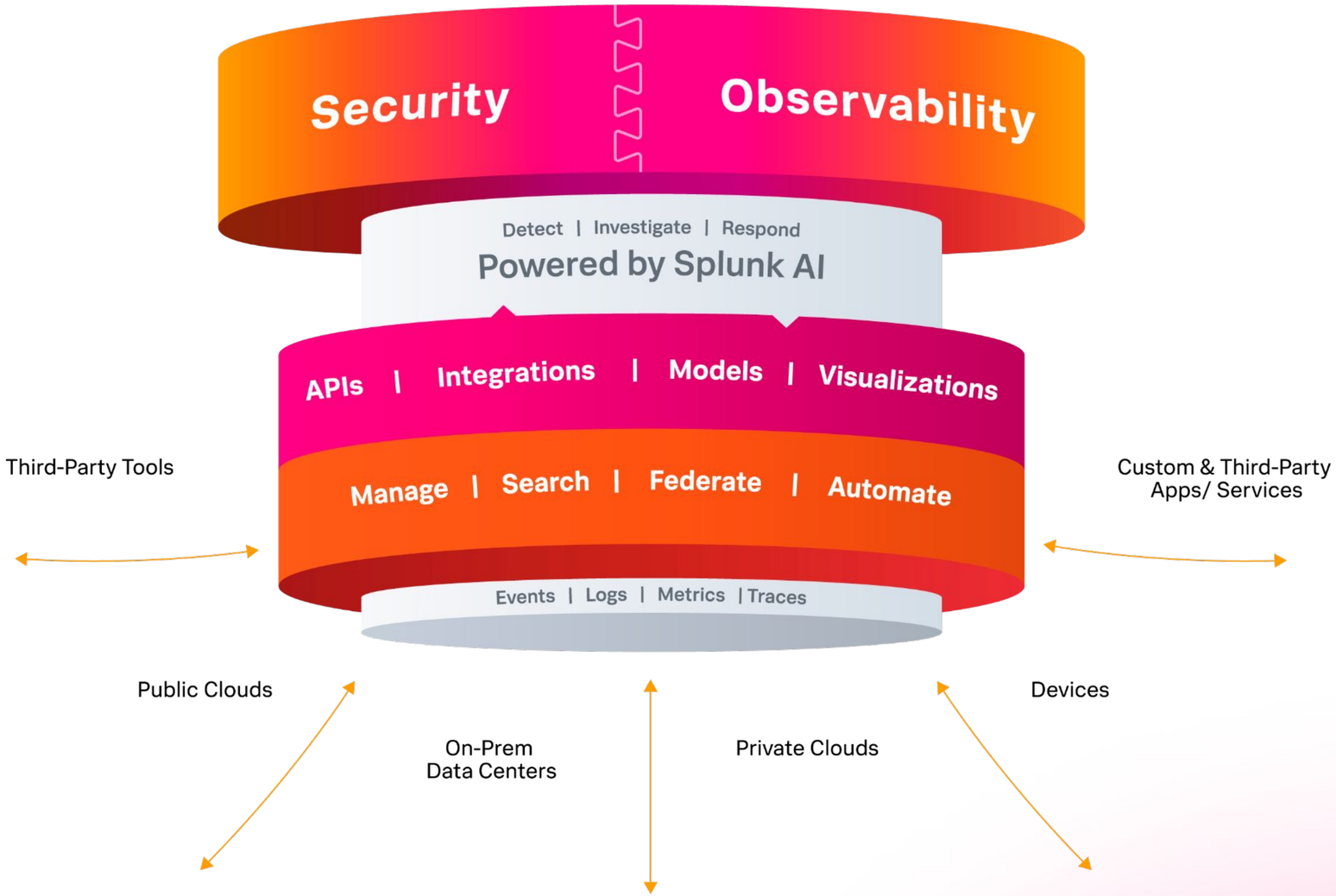
https://www.splunk.com/en_us/form/the-essential-guide-to-security.html

Machine Learning for Security with Splunk



splunk>

Enterprise Security and Observability Platform은 AI에 의해 구동됩니다.



AI와 관련된 앱은 수년전부터 개발해왔습니다.

Splunk has long been committed to helping customers use AI

RESEARCH IN ACTION
Market Leader: Vendor Selection Matrix™ AIOps Solutions

GIGAOM
A Leader in the AIOps Radar for 4 consecutive years

OMDIA
Highest Scoring Product in Selecting an AIOps Solution for 2 years

도메인 별 Use Cases

Security

워크플로우 간소화
머신 러닝 기반 탐지
위험 기반 경고
가이드 응답 조치
이벤트 상관관계 및 경고 소음 감소
예측 분석
이상 탐지
클러스터링

Observability

가능한 근본 원인 분석
이상 및 이상치 탐지
적응형 임계값 설정
경고 상관관계 및 우선순위 지정
보조 복구
사전 장애 예방 지원
권장 응답자

Powered by Splunk AI

Product Overview

Security

- Enterprise Security with Enterprise Security Content Updates (ESCU)
- User Behavior Analytics

Observability

- IT Service Intelligence
- Application Performance Monitoring
- Infrastructure Monitoring
- On Call

제품내의 AI/ML 기능이 포함된 경우

Assistive Intelligence Experiences

- AI Assistant *(Preview)*
- App for Anomaly Detection

Customizable ML

- Machine Learning Toolkit
- App for Data Science and Deep Learning
- Python for Scientific Computing

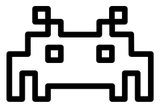
무료로 제공되는 Apps & Tools

The splunk> Platform

- Splunk Cloud Platform
- Splunk Enterprise

Splunk Enterprise Security

보안을 위한 Splunk 머신 러닝 팀의 ML 기반 콘텐츠 업데이트를 확인하세요.



위협 연구

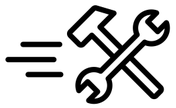
새로운 위협을 식별하고 그 작동 방식을 이해하세요.

데이터 세트 만들기

데이터를 수집하고 Splunk를 사용하여 데이터를 구문 분석하고 위협을 탐지하는 데 사용할 수 있는 패턴을 식별하세요.

머신 러닝 기반 탐지 구축

예측 또는 결정을 내리기 위해 데이터를 기반으로 모델을 구축하고, 시스템이 데이터에서 학습하고, 패턴을 식별하고, 사람의 개입을 최소화하면서 결정을 내릴 수 있도록 지원하며, 위협과 관련된 특정 활동을 식별하도록 설계된 규칙 또는 쿼리를 만들 수 있습니다.



테스트 탐지

공격자 행동을 시뮬레이션하는 데이터 세트에 대해 쿼리를 실행하여 정확도를 개선하고 오탐을 줄입니다.

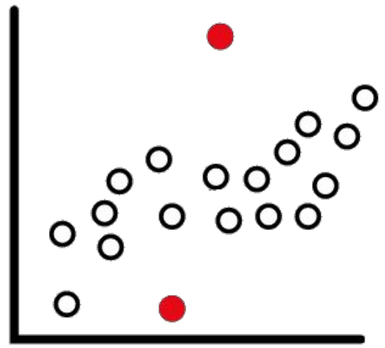
릴리스

Splunk 고객에게 새로운 위협에 대해 시기적절하고 효과적인 보호를 제공하는 패키지 탐지 기능

ML-Powered Detections for Security

데이터 깊숙이 숨어 있는 잘 알려지지 않은 위협 찾기

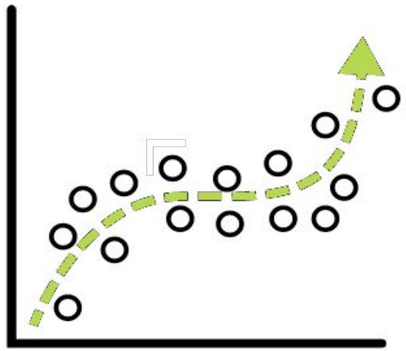
이상 징후 탐지



과거 행동과의 편차

리소스 사용률
오류율 편차
액세스 패턴 기준선

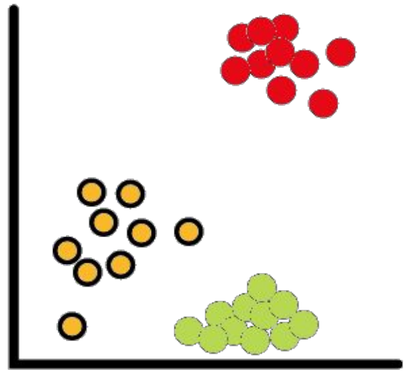
예측 분석



미래 상태 예측
분류/회귀

스토리지 요구 사항 예측
장애로 이어지는 패턴 식별

클러스터링

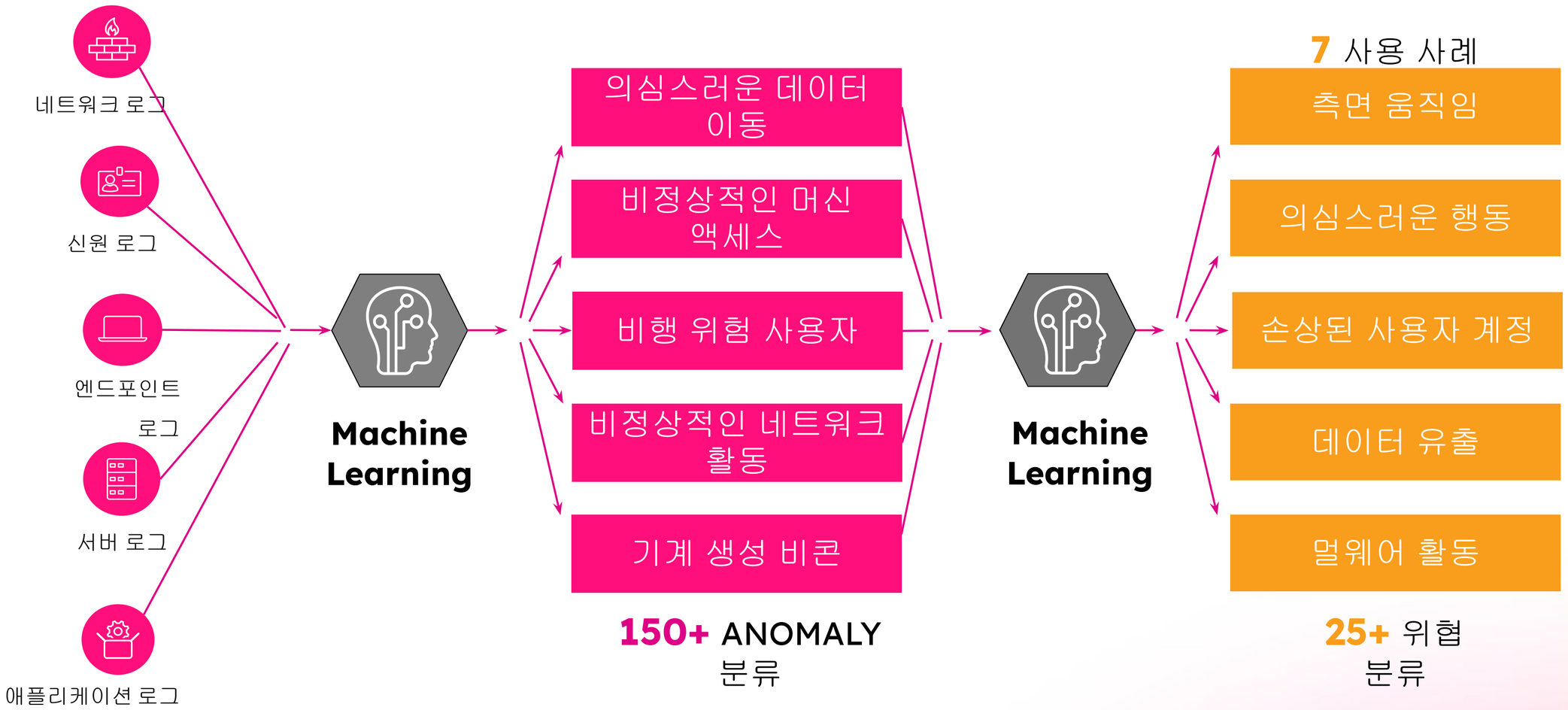


행동 분석

트래픽 식별
행동 분류

User Behavior Analytics

머신 러닝을 사용하여 위협 및 비정상적인 행동 탐지



Splunk AI Assistant

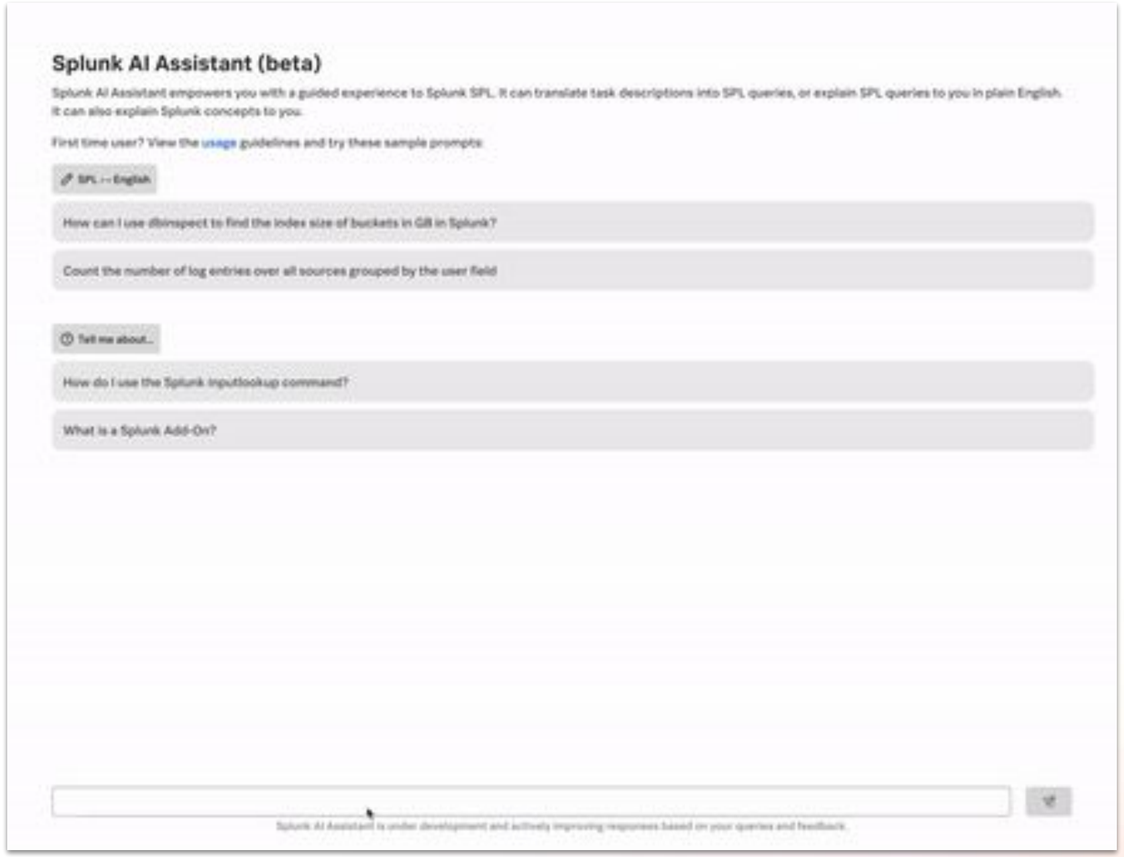
Beta .conf23

생성형 **AI**를 사용하여 더 많은 사용자에게 권한을 부여하고 질문에 더 빠르게 답변하세요.

원하는 쿼리에 대한 설명을 영어로 간단히 작성하고 요청을 SPL 아이디어로 번역할 수 있는 보조 지능 채팅 봇 경험!

다음과 같은 대화형 토론을 할 수 있습니다.

- 자연어 프롬프트를 기반으로 사용자의 질문에 답하는 **SPL** 쿼리 작성하기
- 주어진 **SPL** 쿼리를 평이한 용어로 설명하기



Splunk Enterprise 9.1, Splunk Cloud Platform

Splunk App for Anomaly Detection 1.1.2

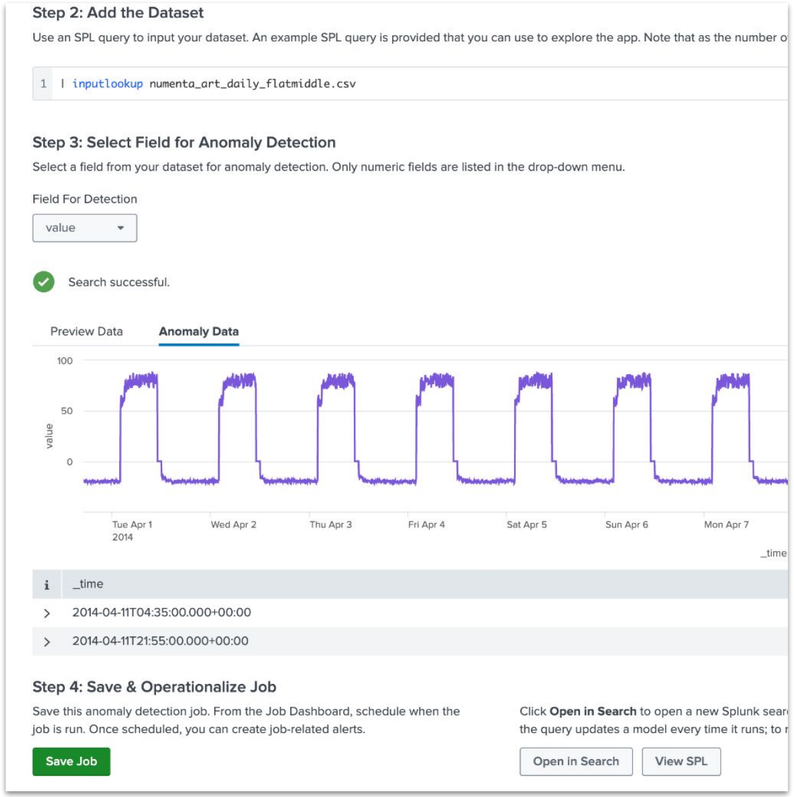
몇 번의 클릭만으로 이상치를 찾아보세요!
https://splunkbase.splunk.com/app/6843

초보자 친화적
복잡한 **SPL** 쿼리, 매개변수 튜닝, 통계 지식이 필요 없습니다.

빠르고 간편함
앱은 몇 번의 클릭만으로 이상 징후를 감지하므로 시행착오가 필요 없습니다.

정확성 보장
상태 확인 진단을 통해 사용자의 데이터 세트가 앱의 알고리즘으로 이상 징후를 탐지하기에 적합한지 여부를 결정합니다.

엔드투엔드 운영 워크플로우
정기적으로 실행하고 경보를 생성할 이상 징후 탐지 작업을 생성하세요.



Splunk Enterprise 9.1, Splunk Cloud Platform

Machine Learning Toolkit 5.5

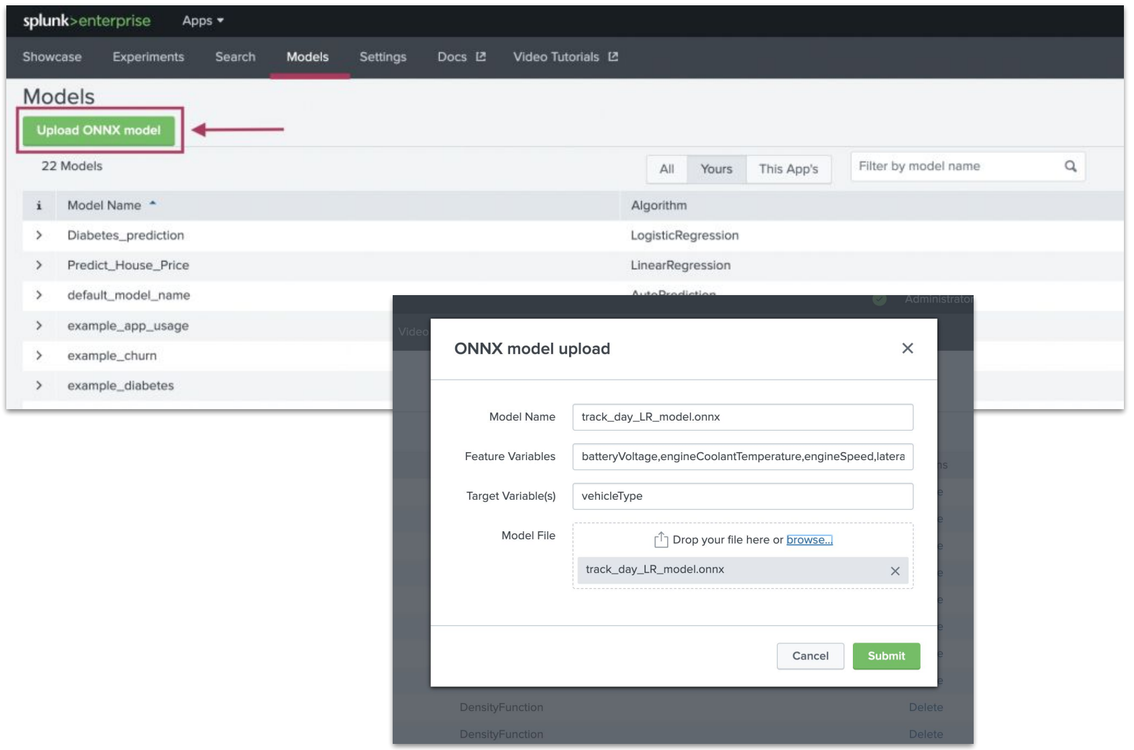
검색 내에서 머신 러닝 사용 사례를 운영하도록 Splunk 확장하기
https://splunkbase.splunk.com/app/2890

모든 수준의 Splunk 사용자를 위한 설계

- **머신 러닝 기반 Splunk 검색:** 검색 내에서 이상 징후 탐색 및 예측과 같은 기술을 적용하여 대시보드 및 인사이트를 강화하세요.
- **쇼케이스 및 실험:** 모델 구축, 테스트 및 배포를 안내하는 간단한 로우코드 환경
- **즉시 확장 가능:** 80개 이상의 내장된 스키킷 학습 알고리즘과 새로운 런타임을 플러그인할 수 있는 API 지원

새로운 업데이트!

- 간단한 UI로 외부에서 사전 학습된 ONNX 모델을 업로드한 다음, 기존 워크플로우를 수정하지 않고도 Splunk 데이터와 함께 모델을 사용할 수 있습니다.
- 다변량 이상값 탐지를 위한 새 알고리즘으로 사용자 이상 징후 탐지 기능 확장



Splunk Enterprise 9.1, Splunk Cloud Platform

Splunk App for Data Science and Deep Learning 5.1

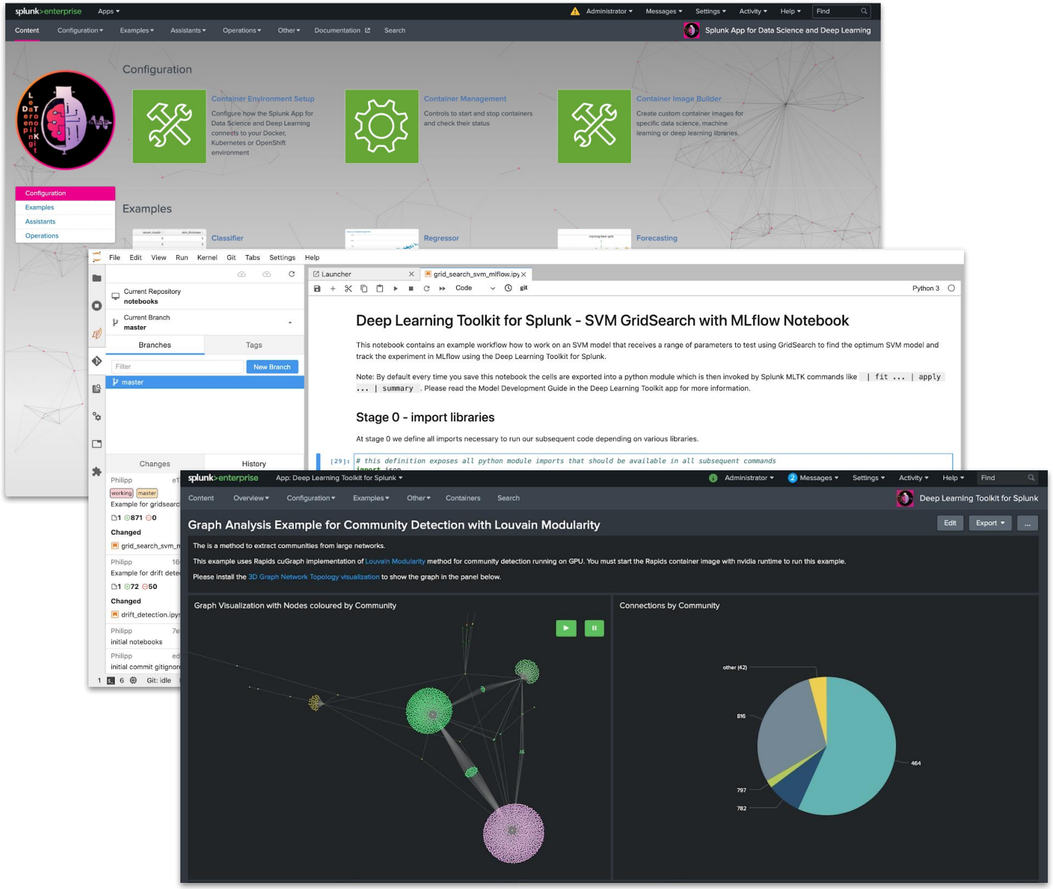
고급 맞춤형 **AI/ML** 사용 사례
https://splunkbase.splunk.com/app/4607

데이터 과학자를 위해 구축

- **35개 이상의 코드 예제:** 데이터 과학 및 딥 러닝 프레임워크의 가이드 모델 구축, 테스트 및 배포
- **컨테이너 관리:** 확장성 및 리소스 최적화(예: **CPU** 및 **GPU**)를 위해 모델을 제작할 수 있습니다.
- **최첨단 AI 프레임워크 및 도구:** Jupyter Lab, MLflow, PyTorch, TensorFlow, SpaCy, DASK, Rapids, Spark, …
- **유연한 배포 및 오픈 소스:** 온프레미스, 하이브리드 또는 클라우드에 배포하세요. 사용자 지정을 위한 **Github** 리포지토리.

버전 5.1의 새로운 업데이트!

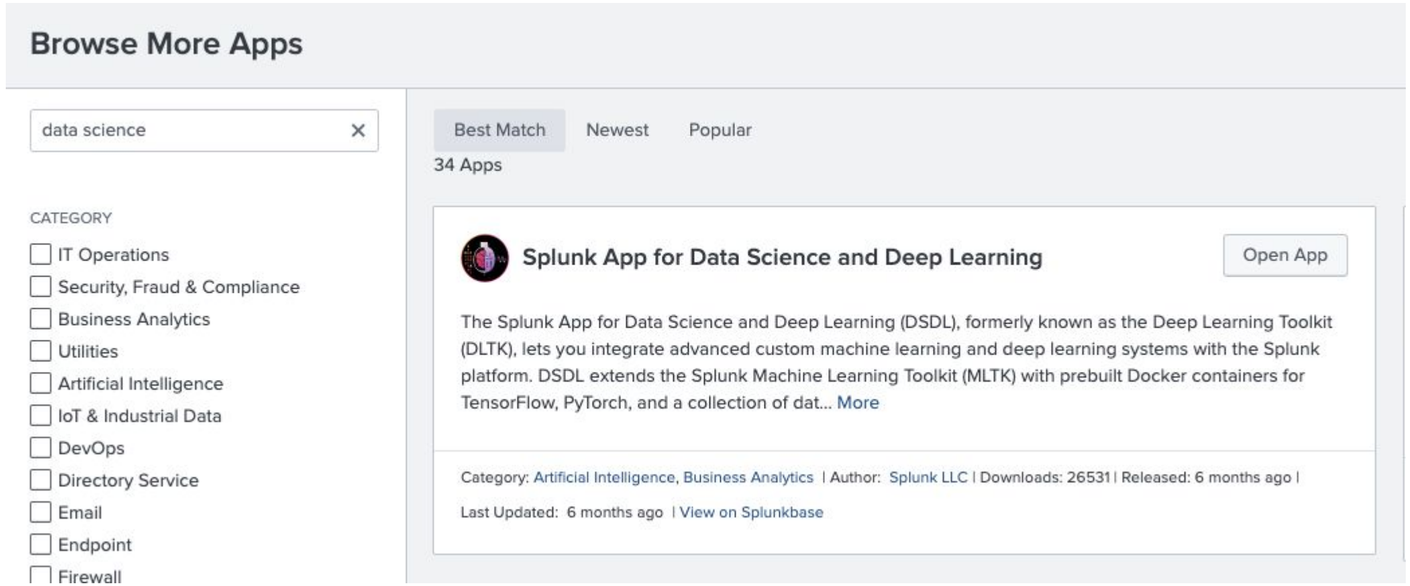
- 텍스트 요약 및 텍스트 분류 사용 사례를 위한 모델을 구축하고 훈련하기 위해 **LLM**을 활용하는 두 가지 **AI** 어시스턴트
- 도메인별 데이터에 맞게 사용자 지정 가능

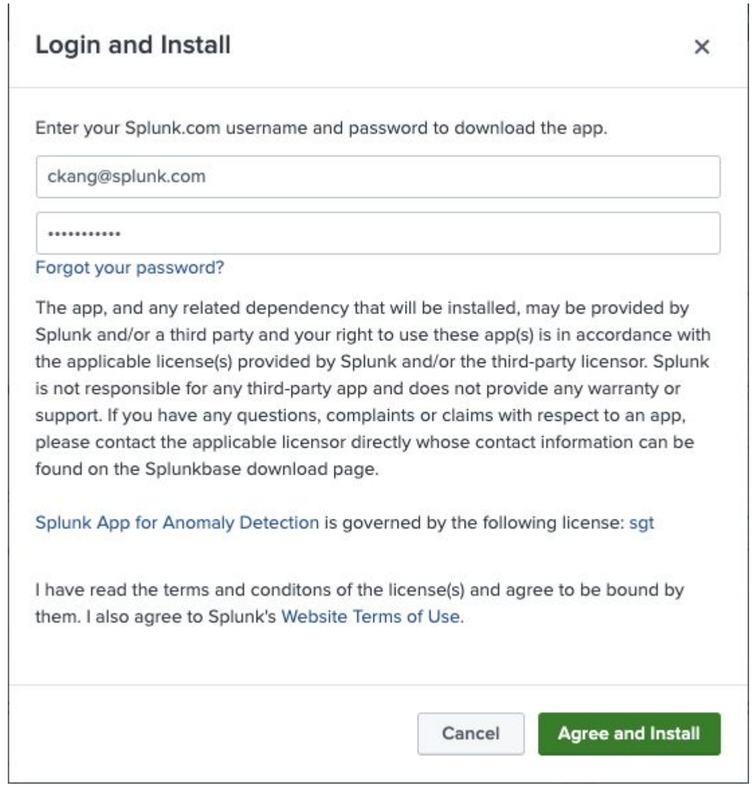


Splunk Enterprise 9.1, Splunk Cloud Platform

Splunk App for Data Science and Deep Learning
설치하기

1. Apps > Browse more apps > `**data science**` 검색 > Install 클릭 > login에 Splunk 가입시 사용한 username/password 기입





Python for Scientific Computing

AI 전용 라이브러리로 Splunk Platform Python 런타임 확장

지원되는 광범위한 오픈 소스 Python 라이브러리를 사용하여 Splunk Platform에서 복잡한 AI/ML 기반 분석 을 실행하세요.

Where Are We Going with AI?

Unified experience across Splunk with a scalable backend to power them

MLTK 기능에서 Splunk의 데이터에 대한 엔드투엔드 ML 안내로 전환하고 있습니다.

더욱 신뢰할 수 있는 Splunk 고유의 생성 AI

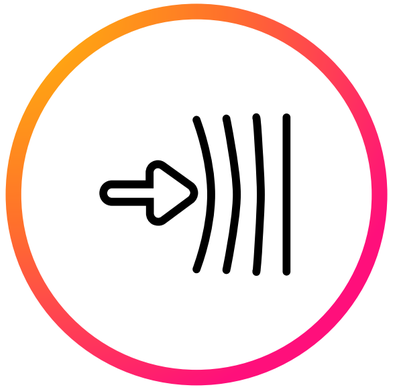
Splunk AI Assistant의 생성 AI를 개선하고 다른 사용 사례 및 제품으로 확장하기

추가 임베디드 AI

Splunk 제품에서 사용자의 일상적인 워크플로에 더 많은 AI를 통합하고 더 많은 지원 환경을 구축하세요.

규모에 맞는 ML 실행

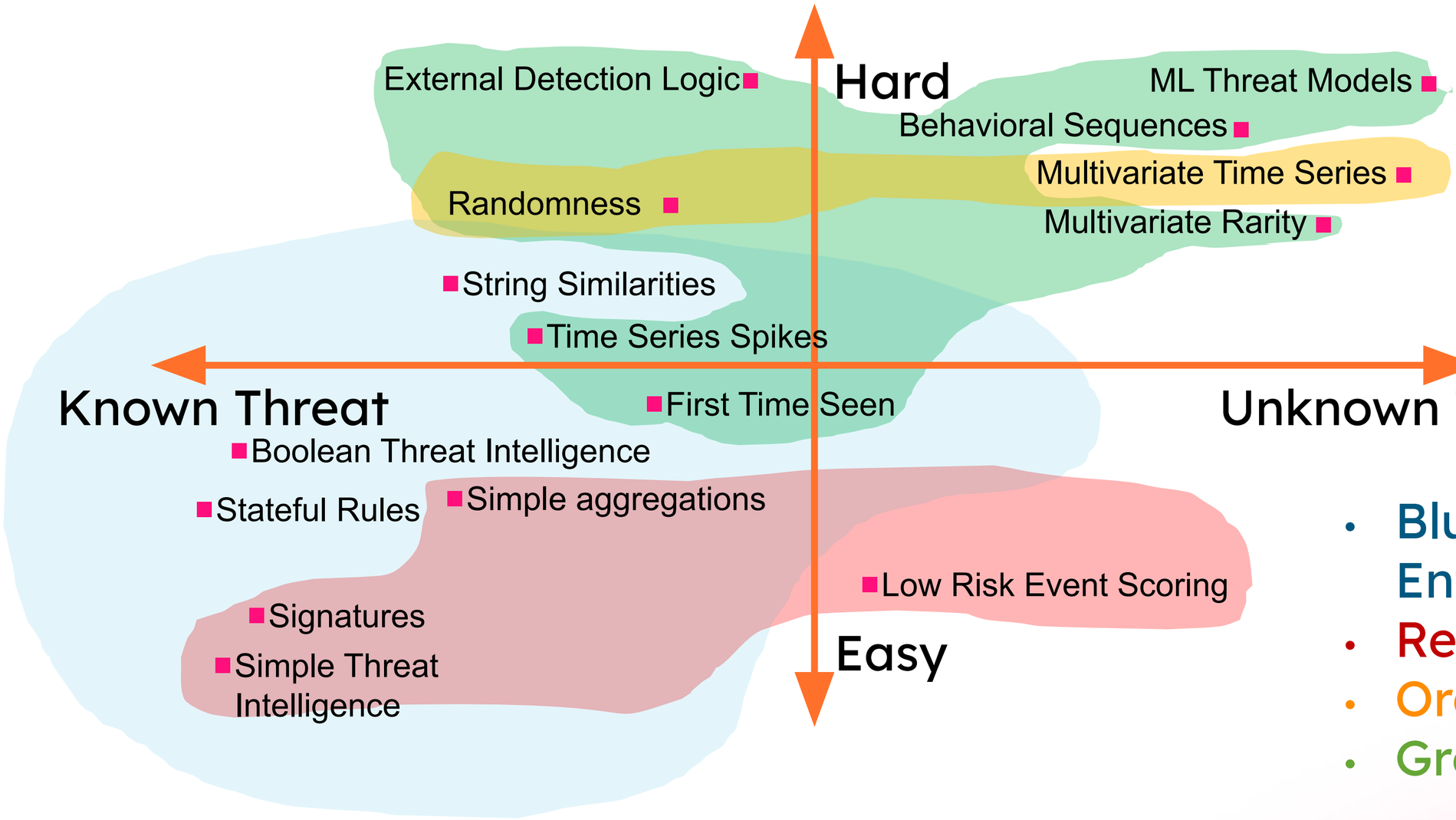
대규모 ML 모델 학습 및 배포를 지원하는 강력한 런타임 개발



Splunk 개발자를 위한 확장 가능한 ML

개발자가 ML 기반 환경을 만들 수 있도록 SDK를 구축하세요.

Detective Controls by Difficulty



- Blue - Splunk Enterprise
- Red - Splunk ES
- Orange - Splunk MLTK
- Green – Splunk UBA

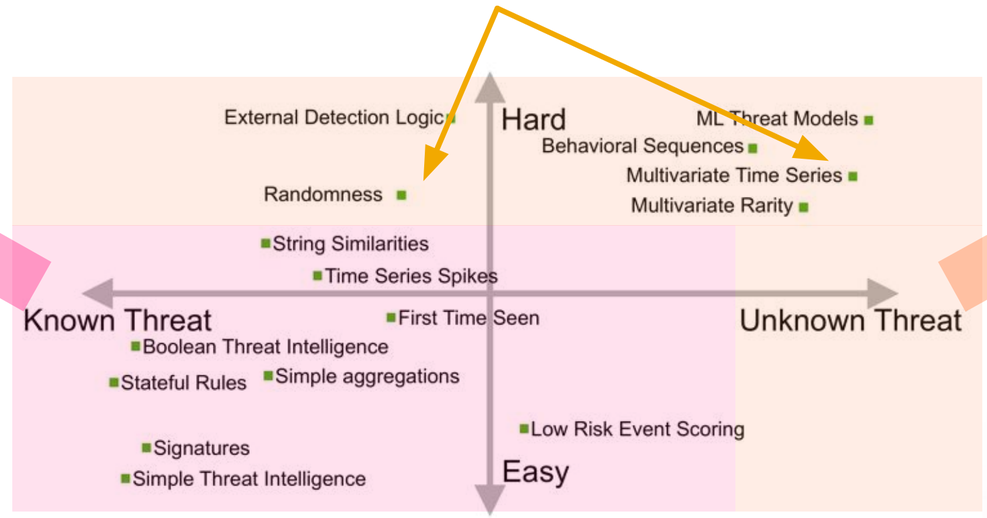
Splunk Products

Splunk User Behavior Analytics

- Extensive difficult-to-build detections
- Customization difficult (Java or Scala)
- Heavy ML to detect unknown threats

Splunk Machine Learning Toolkit

- Customization leverages Python
- Some unique capabilities



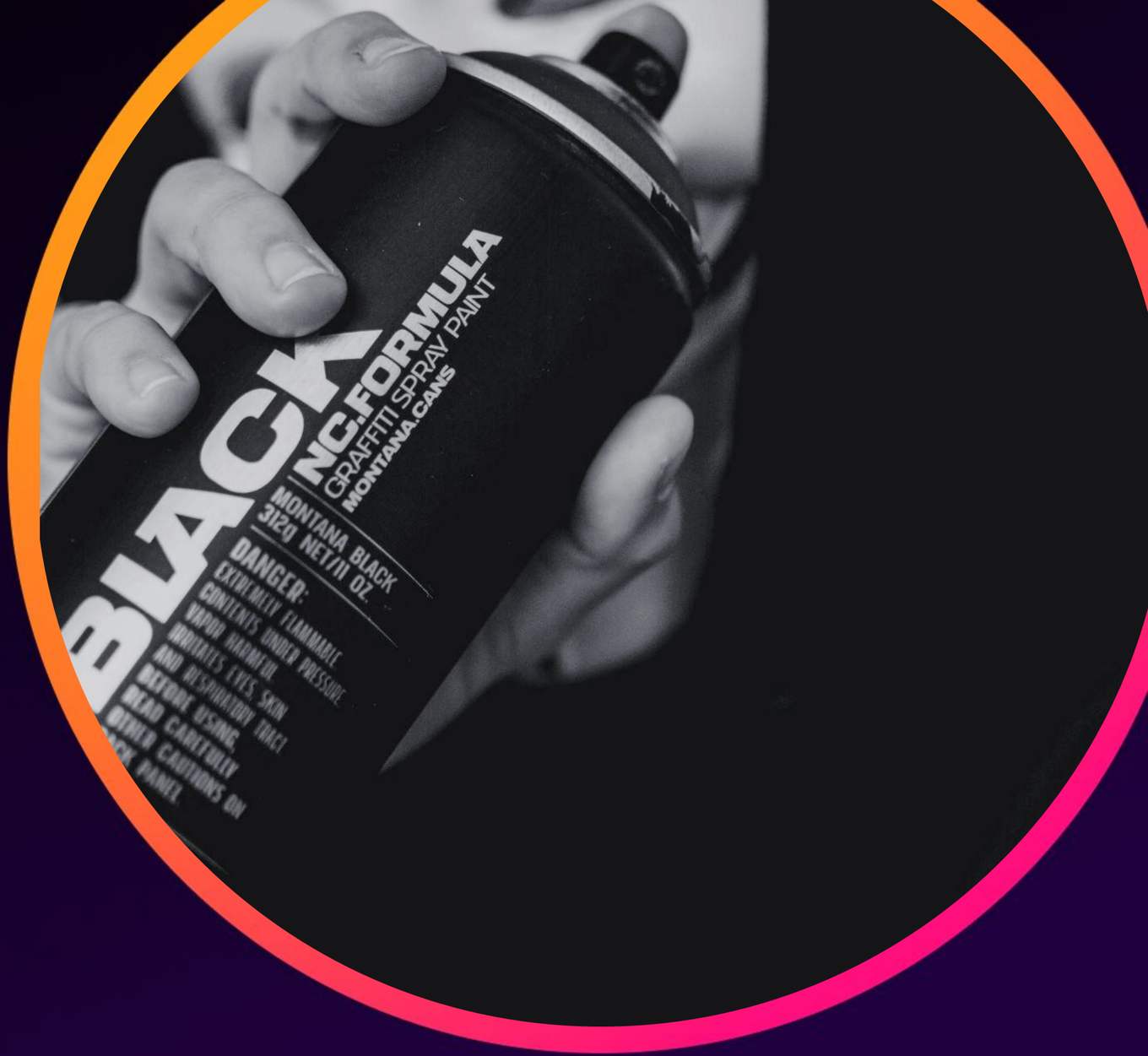
Splunk Enterprise + Enterprise Security

- Best customization
- Easily understood
- Fast iteration
- Focused on easier detections for known threats

Hands-on labs

Lab 1

Detect Password Spraying Attack- Credential Access



Detect Password Spraying Attack

Credential Access

Lab 1

Detect Password Spraying Attack- Credential Access

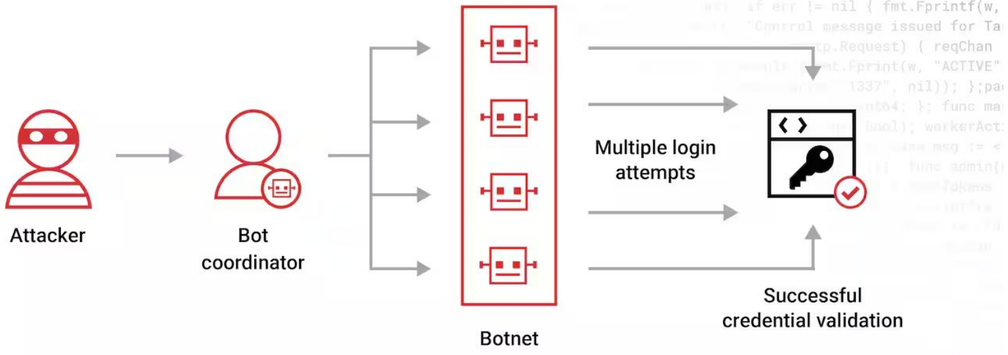
Password Spraying이란 무엇일까요?

MITRE ATT&CK Framework 정의하는 공격기법 T1110.003(Password spraying)은, 특정 사용자 계정에 대해 여러 비밀번호를 시도하는 brute-force attack과는 다른 공격 기법입니다.
공격자는 여러 사용자에 대해 같은 비밀번호를 시도하여 계정 잠기지 않게 공격을 수행합니다. 그렇기 때문에 공격자는 계정 잠금을 피해 공격을 수행합니다.

그렇기에 이 기법은 적절한 탐지 방법 없이는 탐지하기 어렵습니다. 이러한 공격 기법은 단일 로그인(SSO) 및 클라우드 기반 어플리케이션에 사용하는 인증을 표적으로 삼는데요,

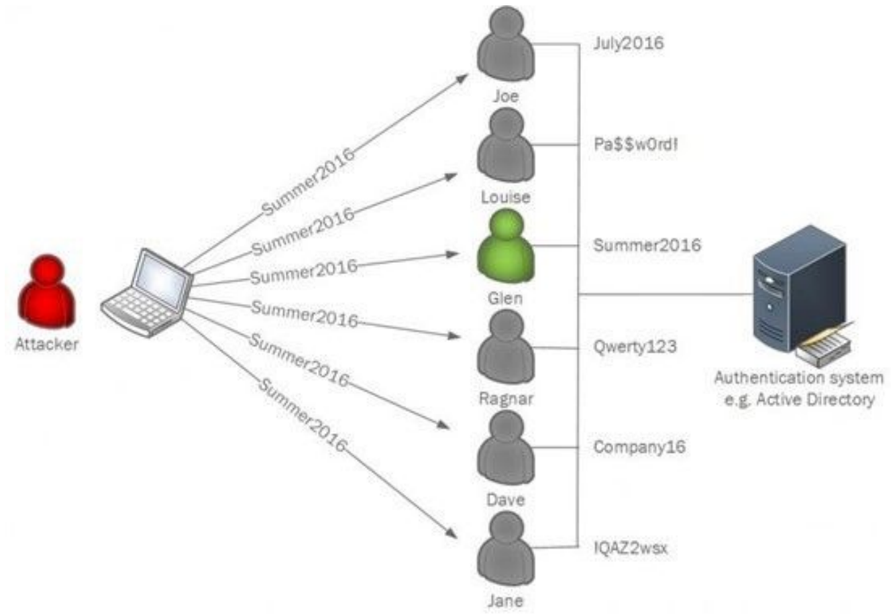
공격자가 초기 접근 권한을 얻으면 추후에 조직의 중요한 데이터에 접근할 수 있습니다.

Brutu force VS Password Sparying



How a brute-force attack works

Akamai



간단하게 `stats` 사용

Method 1

Detect Password Spraying Attack

Credential Access

Lab 1

First Technique#1: 간단한 stats

Password Spraying Attacks을 탐지하기 위해서

이러한 공격을 어떻게 탐지할까요?

Tips: Windows Event Logs – 다음 중 디바이스 로그인 시도 실패를 나타내는 이벤트 코드는? 원격 인증 시도를 나타내는 이벤트는 무엇입니까? (선택사항) **Process** 이름 당 혹은 컴퓨터 명을 통해서도 필터링을 할 수 있습니다.

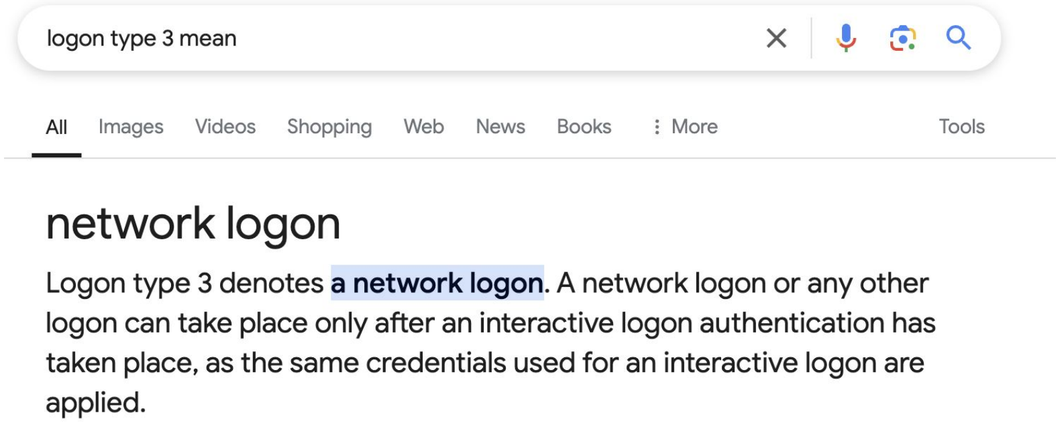
첫번째 **lab**을 수행하기 위해서 다음의 명령어를 사용해 보세요. -`eval`, `stats`, `timechart`, `evenstats`- 또한 표준편차를 사용하는걸 잊지 마세요!

Tips - 1

1. **Tip 1:** index="main" sourcetype="XmlWinEventLog_ws"
2. **Tip 2**: Password Spraying을 의미하는 EventCode=4625

> In default environments, LDAP and Kerberos connection attempts are less likely to trigger events over SMB, which creates Windows "logon failure" event ID 4625.

3. **Tip 3**: 인터넷 상으로 시도된 LogonType=3



4. **Tip 4**: 시간을 2m단위로 그룹화

```
index="main" sourcetype="XmlWinEventLog_ws" EventCode=4625 LogonType=3
| bucket span=2m _time
```

Tips - 2

Tip 1: 사용자 이름(TargetUserName)을 dc와 values를 통해 리스트화

a. dc(field): field의 고유값을 반환
b. values(field): field의 고유값을 통해 배열로 반환

Tip 2: _time, IpAddress, LogonType, dvc로 그룹화

Tip 3: 각 계정(unique_accounts)에 대한 평균과 표준편차 구하기

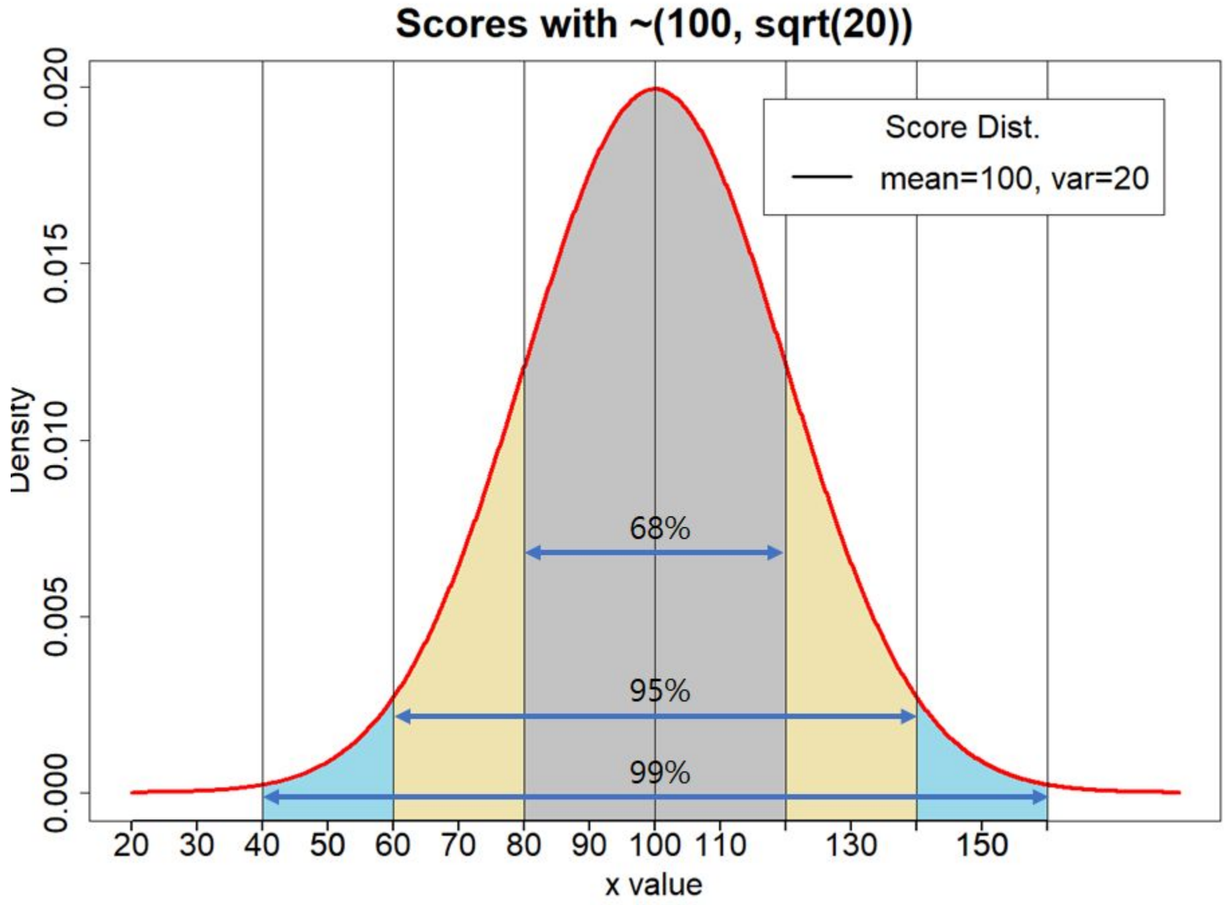
a. avg(field)
b. stdev(field)

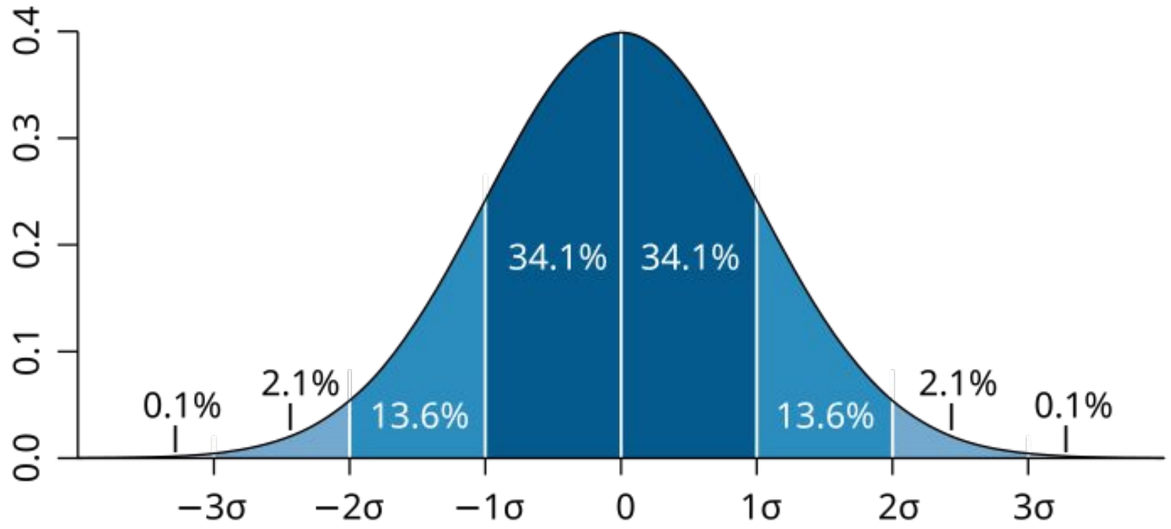
Tip 4: 2σ(시그마)의 기준값 구하기

a. | eval upperBound=(comp_avg+comp_std*2)

```
| stats dc(TargetUserName) as unique_accounts values(TargetUserName) as
  tried_accounts by _time, IpAddress, LogonType, dvc
| eventstats avg(unique_accounts) as comp_avg, stdev(unique_accounts) as comp_std
  by IpAddress, LogonType, dvc
| eval upperBound=(comp_avg+comp_std*2)
```

여기서 잠깐? 표준편차와 평균을 사용하는 이유는?





평균(avg)

Tips - 3

1. **Tip 1:** 동일한 조건에서 여러 계정에 대해 로그인을 시도한 계정을 식별하는 field는 **unique_accounts**
2. **Tip 2**: unique_accounts가 2σ(시그마) 이상이거나 6번 초과일 경우 isOutlier로 지정
3. **Tip 3**: isOutlier인 값들만 보기

```
| eval isOutlier=if(unique_accounts > 6 AND unique_accounts >= upperBound, 1, 0)
| search isOutlier=1
```

First Technique#1: Simple stats

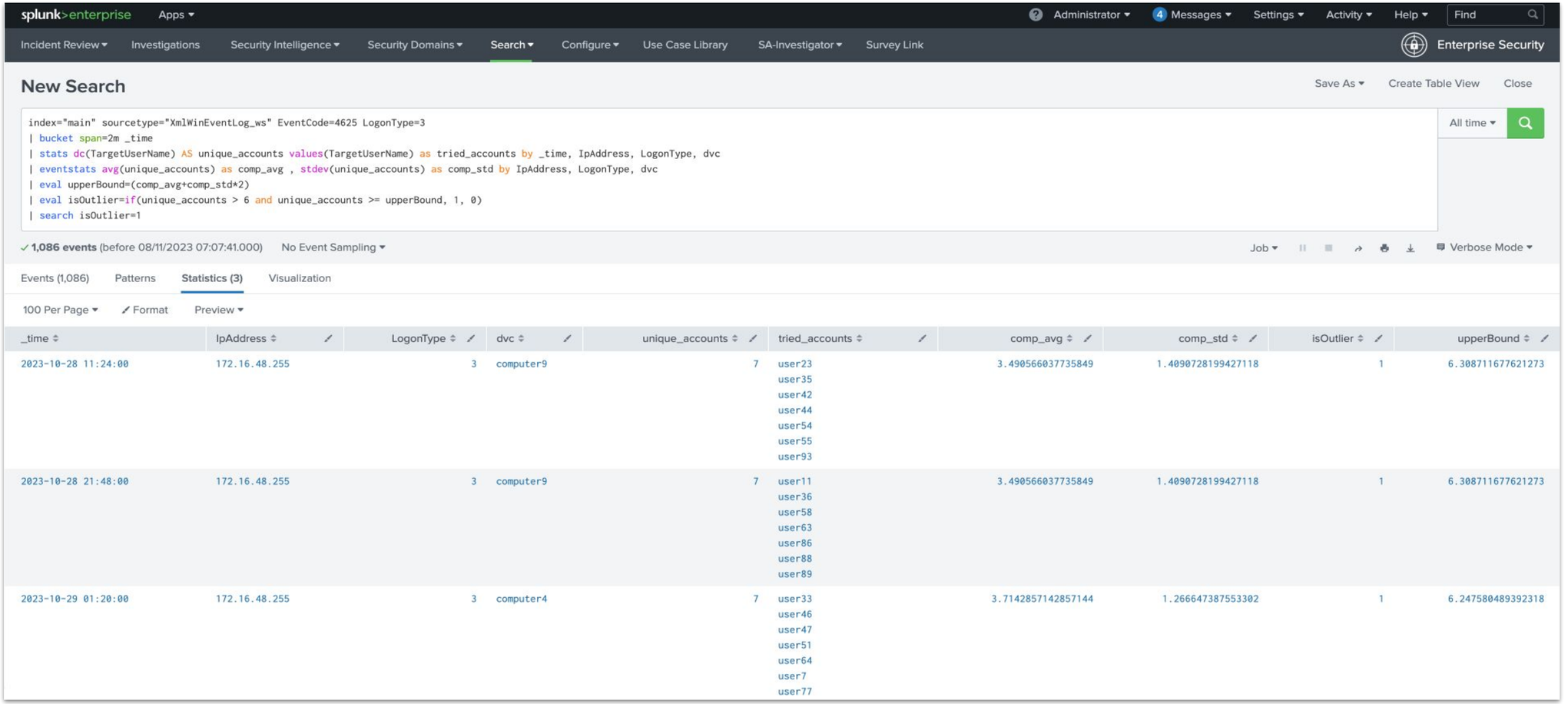
Detect Remote Password Spraying Attacks

```
index="main" sourcetype="XmlWinEventLog_ws" EventCode=4625 LogonType=3
| bucket span=2m _time
| stats dc(TargetUserName) as unique_accounts values(TargetUserName) as
  tried_accounts by _time, IpAddress, LogonType, dvc
| eventstats avg(unique_accounts) as comp_avg, stdev(unique_accounts) as comp_std
  by IpAddress, LogonType, dvc
| eval upperBound=(comp_avg+comp_std*2)
| eval isOutlier=if(unique_accounts > 6 AND unique_accounts >= upperBound, 1, 0)
| search isOutlier=1
```

Finding and Removing Outliers: https://splk.it/3s0Pig5

First Technique#1: Simple stats

Detect Remote Password Spraying attacks



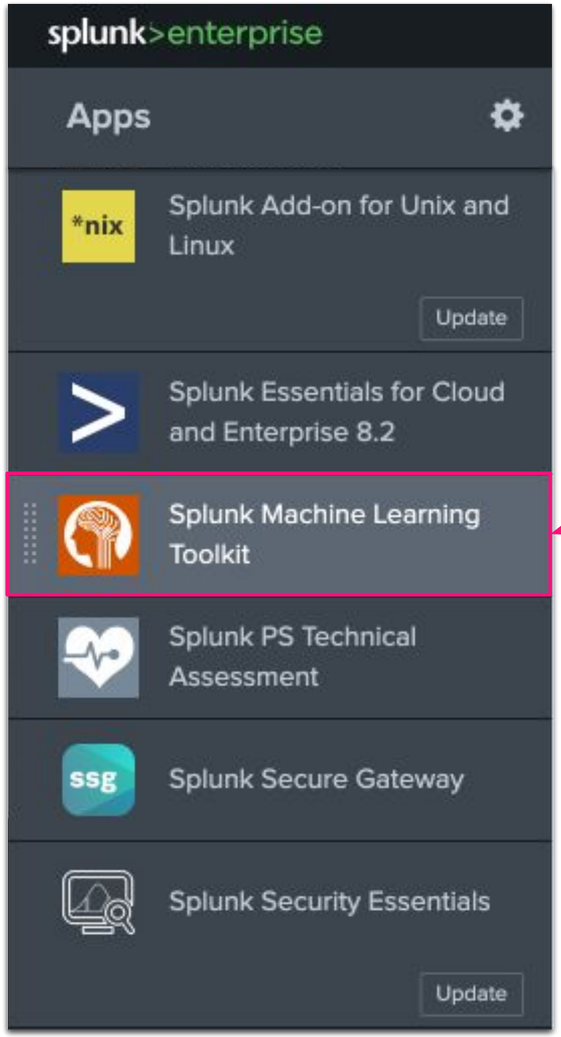
Finding and Removing Outliers: https://splk.it/3s0Pig5

Using MLTK

Method 2

MLTK:

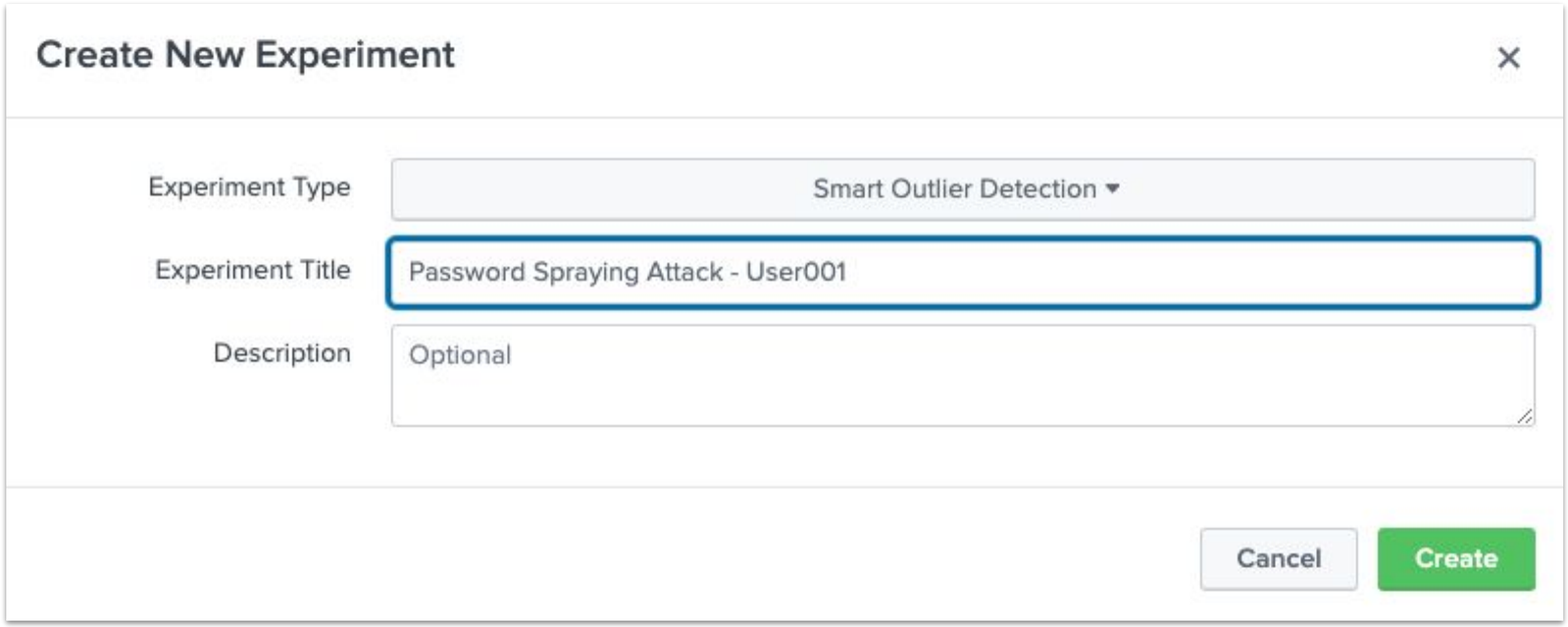
Apps 메뉴에서 **MLTK** 앱을 선택하세요:



Open the MLTK app

Second Technique#2: Using MLTK

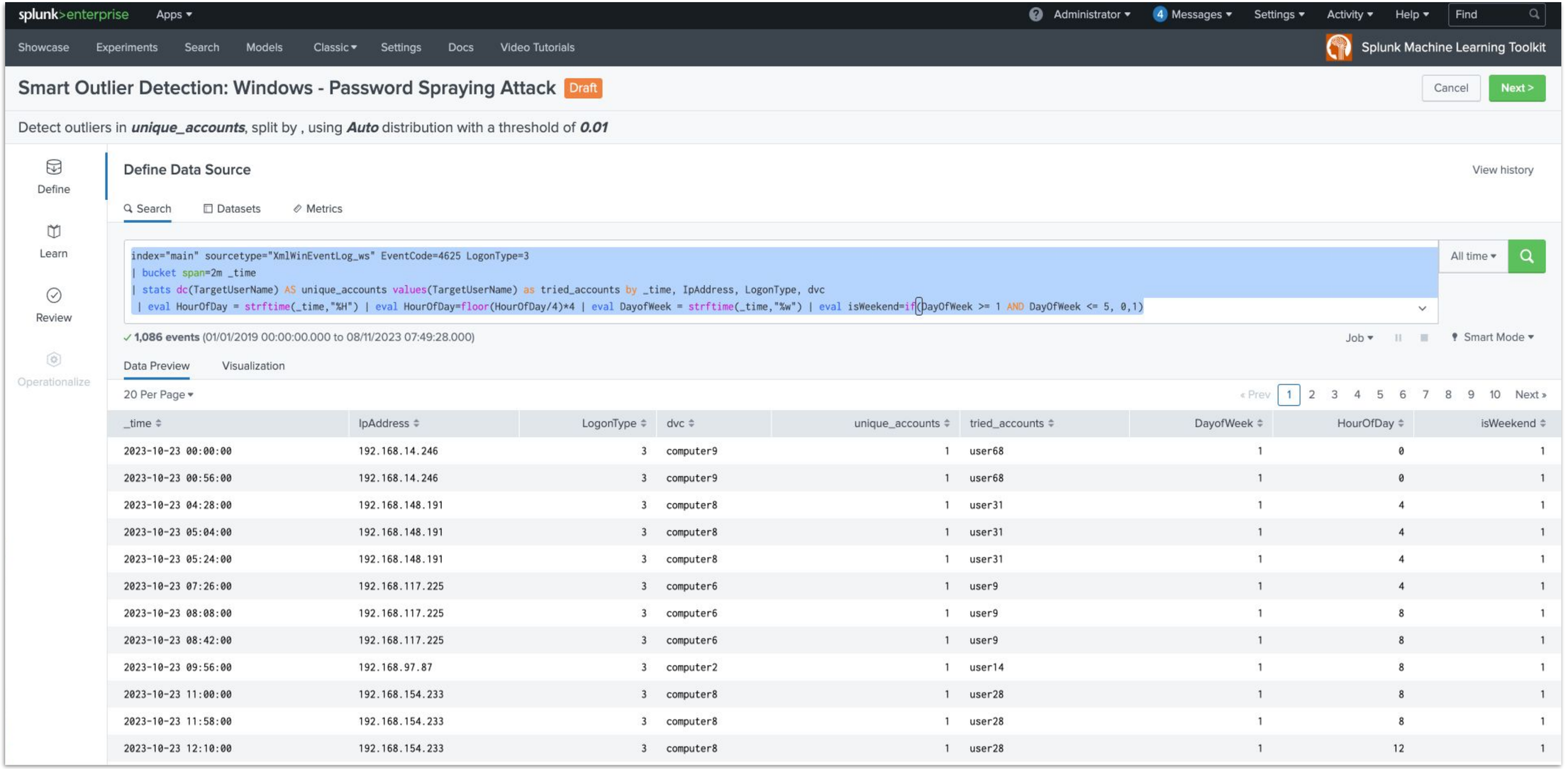
1. Machine Learning Toolkit 앱으로 이동
2. Experiment(실험) 선택
3. “Smart Outlier Detection” 선택 후 이름을 다음과 같이 변경
   - Password Spraying Attack - [사용자명]



Search에 다음과 같이 입력

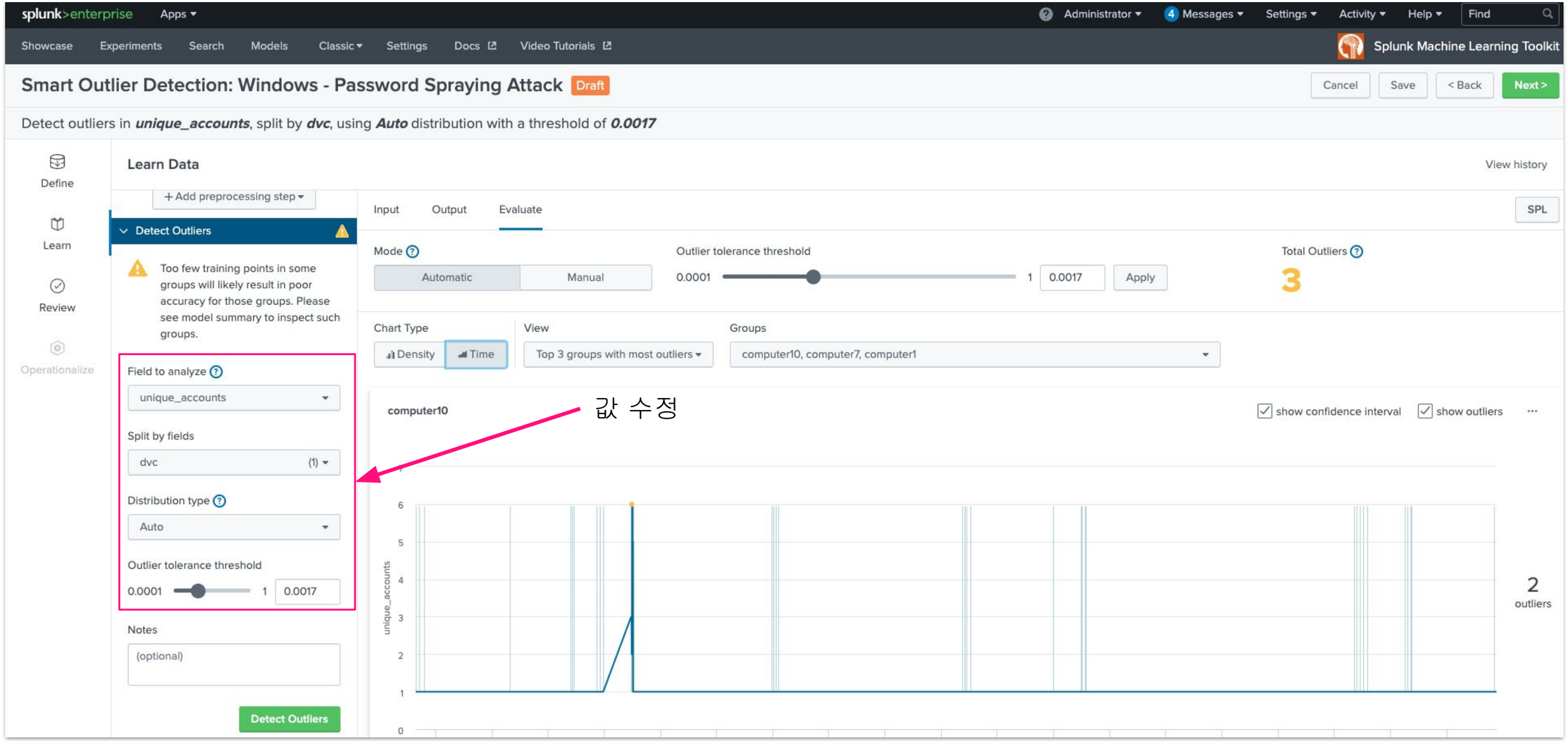
```
index="main" sourcetype="XmlWinEventLog_ws" EventCode=4625 LogonType=3
| bucket span=2m _time
| stats dc(TargetUserName) as unique_accounts values(TargetUserName) as
  tried_accounts by _time, IpAddress, LogonType, dvc
| eventstats avg(unique_accounts) as comp_avg, stdev(unique_accounts) as comp_std
  by IpAddress, LogonType, dvc
| eval HourOfDay = strftime(_time,"%H")
| eval HourOfDay = floor(HourOfDay/4)*4
| eval DayofWeek = strftime(_time,"%w")
| eval isWeekend = if(DayOfWeek >= 1 AND DayOfWeek <= 5, 0,1)
```

Second Technique#2: Using MLTK

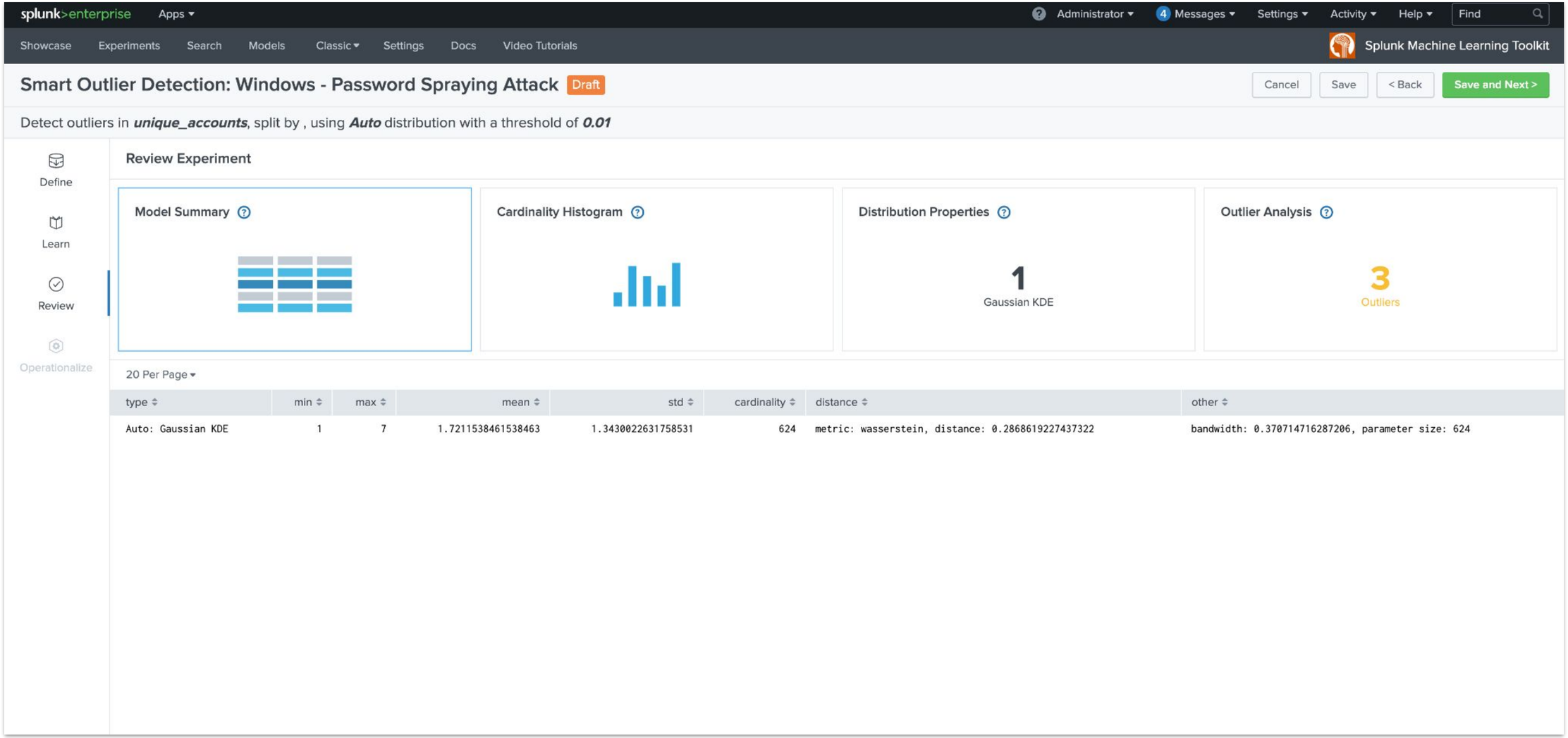


_time	IpAddress	LogonType	dvc	unique_accounts	tried_accounts	DayofWeek	HourOfDay	isWeekend
2023-10-23 00:00:00	192.168.14.246	3	computer9	1	user68	1	0	1
2023-10-23 00:56:00	192.168.14.246	3	computer9	1	user68	1	0	1
2023-10-23 04:28:00	192.168.148.191	3	computer8	1	user31	1	4	1
2023-10-23 05:04:00	192.168.148.191	3	computer8	1	user31	1	4	1
2023-10-23 05:24:00	192.168.148.191	3	computer8	1	user31	1	4	1
2023-10-23 07:26:00	192.168.117.225	3	computer6	1	user9	1	4	1
2023-10-23 08:08:00	192.168.117.225	3	computer6	1	user9	1	8	1
2023-10-23 08:42:00	192.168.117.225	3	computer6	1	user9	1	8	1
2023-10-23 09:56:00	192.168.97.87	3	computer2	1	user14	1	8	1
2023-10-23 11:00:00	192.168.154.233	3	computer8	1	user28	1	8	1
2023-10-23 11:58:00	192.168.154.233	3	computer8	1	user28	1	8	1
2023-10-23 12:10:00	192.168.154.233	3	computer8	1	user28	1	12	1

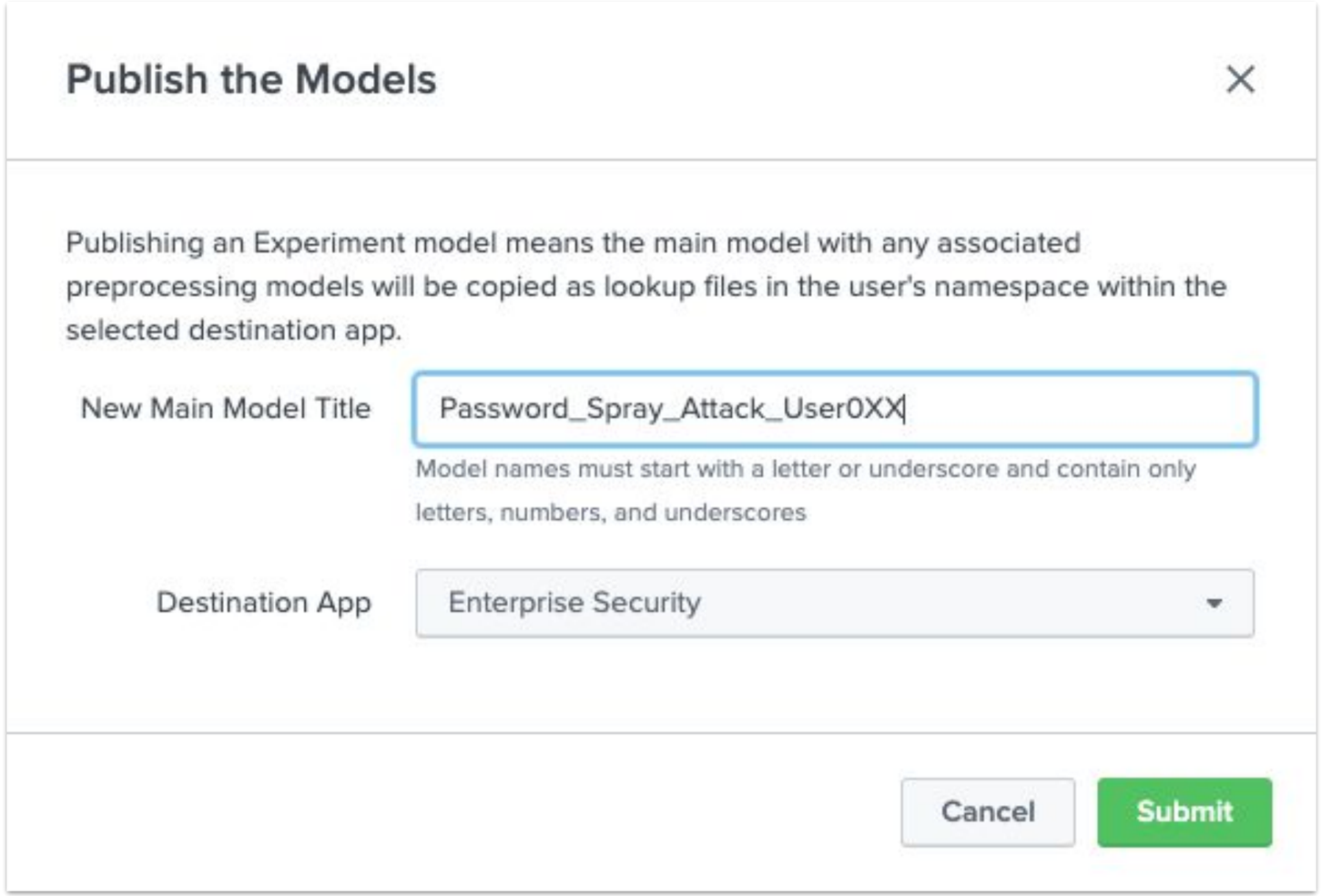
Second Technique#2: Using MLTK



Second Technique#2: Using MLTK



Publish Your Model



20 minutes to complete the lab

Lab 1

Privilege Escalation

Lab 2

Using Simple stats

Method 1

First technique#1: Simple `stats`

권한 상승 이벤트 탐지하기

권한 상승이 된 이벤트의 이상치를 어떻게 탐지할 수 있을까요? 데이터를 먼저 이해해야합니다.

Tips: 권한 상승의 의미가 있는 **EventCode 4648**을 이용하여 **Windows Event Logs**를 살펴포세요.
`stats, sort` 명령어를 사용해서 이벤트의 크기와 **user** 단위를 그룹화

Other stats functions: `stdev()`, `perc99.999()`...

과거 데이터이기에 시간대를 **(08/13/2020 - 09/01/2020)** 로 변경해 주세요!

First technique#1: Simple stats

권한 상승 이벤트 탐지하기

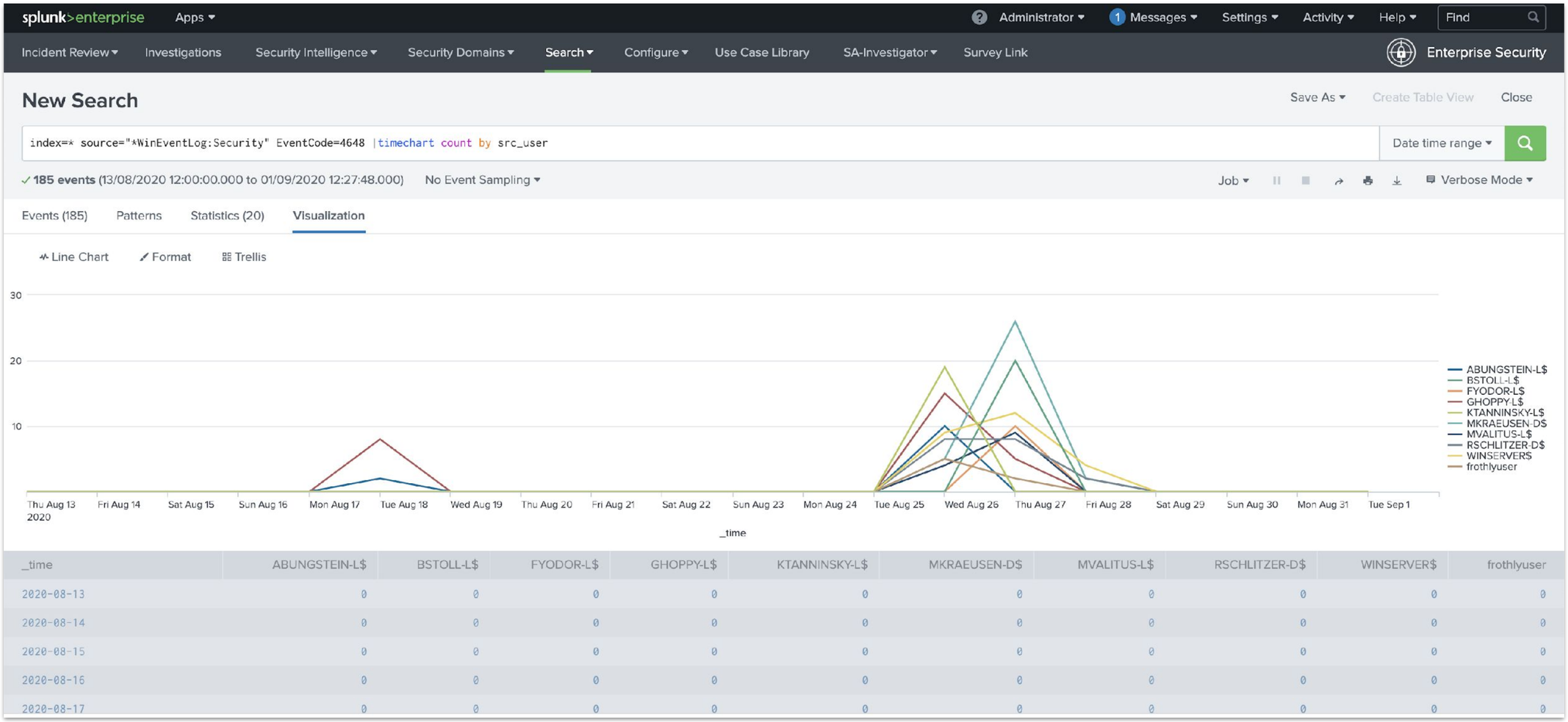
권한 상승이 된 이벤트의 이상치를 어떻게 탐지할 수 있을까요? 데이터를 먼저 이해해야합니다.

Tips: 권한 상승의 의미가 있는 **EventCode 4648**을 이용하여 **Windows Event Logs**를 살펴포세요.
`stats, sort` 명령어를 사용해서 이벤트의 크기와 **user** 단위를 그룹화

```
index="main" source="*WinEventLog:Security" EventCode=4648
| timechart count by src_user
```

First technique#1: Simple stats

Detecting Privileged Escalation Events



_time	ABUNGSTEIN-L$	BSTOLL-L$	FYODOR-L$	GHOPPY-L$	KTANNINSKY-L$	MKRAEUSEN-D$	MVALITUS-L$	RSCHLITZER-D$	WINSERVER$	frothlyuser
2020-08-13	0	0	0	0	0	0	0	0	0	0
2020-08-14	0	0	0	0	0	0	0	0	0	0
2020-08-15	0	0	0	0	0	0	0	0	0	0
2020-08-16	0	0	0	0	0	0	0	0	0	0
2020-08-17	0	0	0	0	0	0	0	0	0	0

First technique#1: Simple stats

권한 상승 이벤트 탐지하기

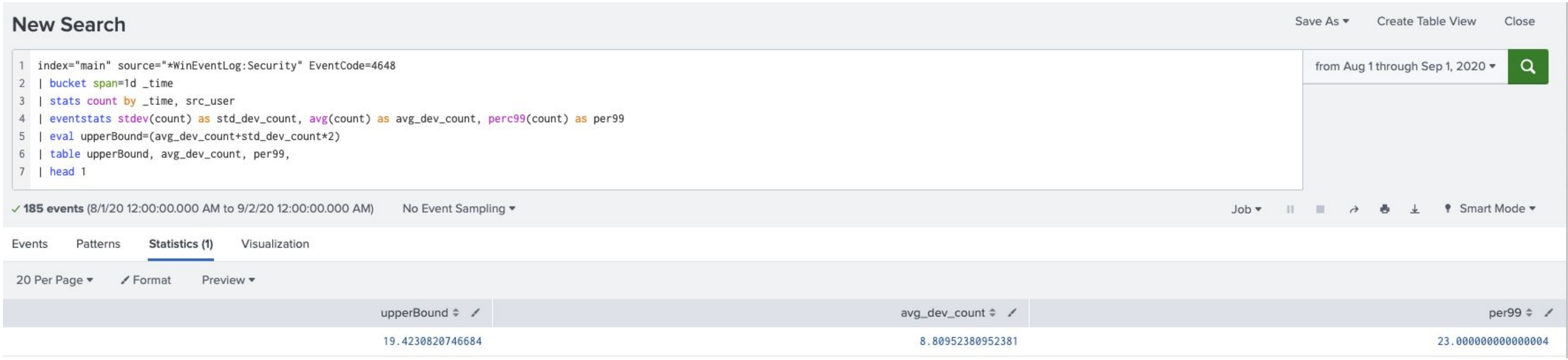
Eventstats, stdev, eval 함수를 사용해 이상치를 찾아봅시다.

함수에 대한 설명은 다음의 링크를 참고하세요!
https://docs.splunk.com/Documentation/SCS/current/SearchReference/Aggregatefunctions#stdev.28.26lt.3Bvalue.26gt.3B.29

물론 이거보다 더 쉬운 방법은 있지만요! ☺️

```
index="main" source="*WinEventLog:Security" EventCode=4648
| bucket span=1d _time
| stats count by _time, src_user
| eventstats stdev(count) as std_dev_count, avg(count) as avg_dev_count,
perc99(count) as per99
| eval upperBound=(avg_dev_count+std_dev_count*2)
| table upperBound, avg_dev_count, per99
| head 1
```

First technique#1: Simple `stats`



First technique#1: Simple stats

권한 상승 이벤트 탐지하기

2σ 밖에 있는 값을 찾아봅시다!

```
index="main" source="*WinEventLog:Security" EventCode=4648
| bucket span=1d _time
| stats count by _time, src_user
| eventstats stdev(count) as std_dev_count, avg(count) as avg_dev_count,
perc99(count) as per99, perc95(count) as per95
| eval upperBound=(avg_dev_count+std_dev_count*2)
| eval isOutlier=if(count >= upperBound, 1, 0)
| search isOutlier=1
```

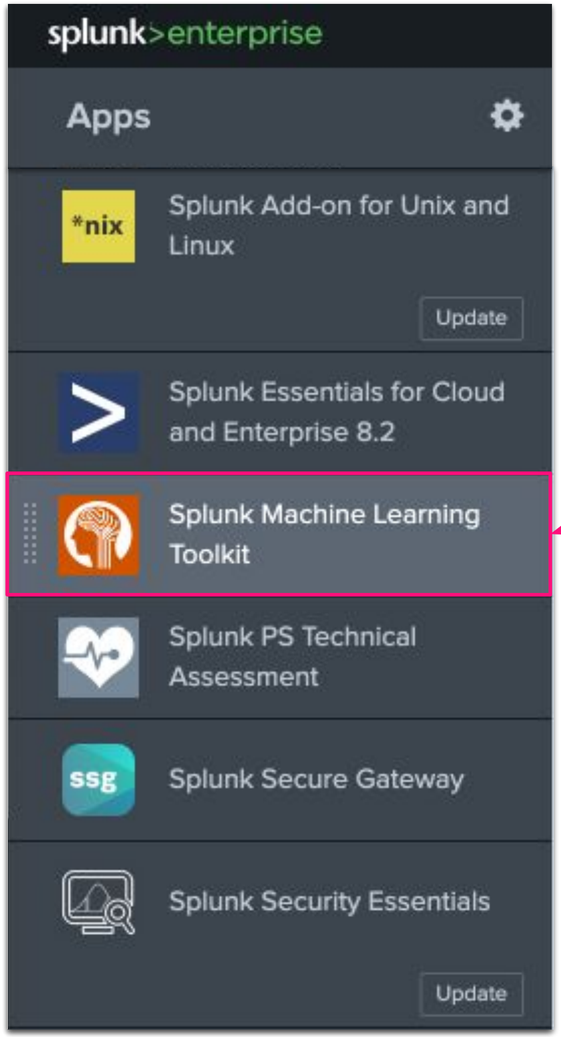
per99 밖에 있는 값을 찾으려면? `count >= upperBound` 를 `count >= per99` 로 변경!

Using MLTK

Method 2

MLTK:

Apps 메뉴에서 **MLTK** 앱을 선택하세요:



Open the MLTK app

MLTK:

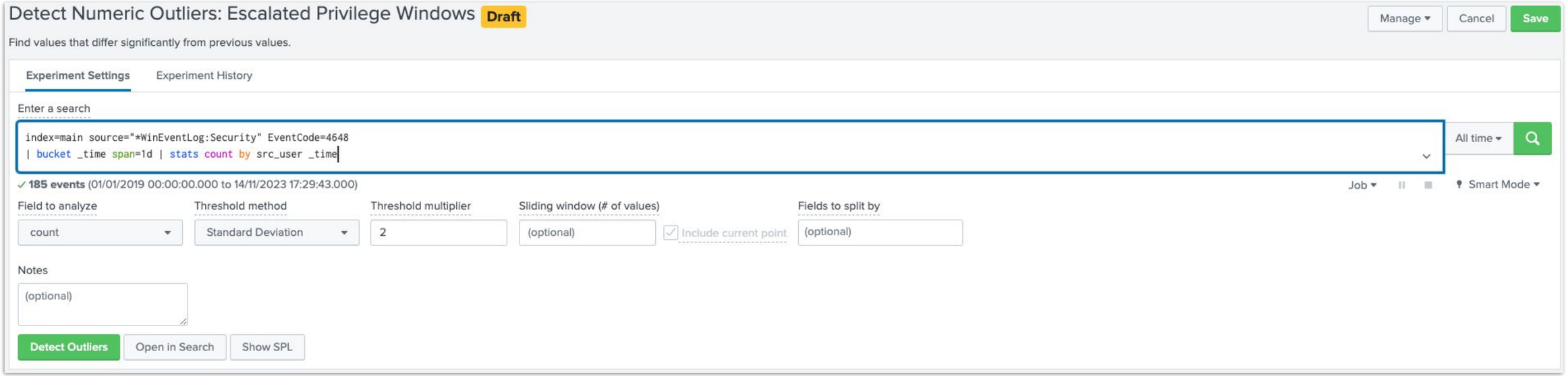
Experiment을 만들어 볼까요?

1. **Experiments**로 이동
2. **Detect Numeric Outliers**선택
3. **New Experiment** 생성
4. **Title** 변경 **(Detecting Privilege Escalation - ch)**
5. 다음의 검색문 실행

```
index=main source="*WinEventLog:Security" EventCode=4648
| bucket _time span=1d | stats count by src_user _time
```

6. **Field to analyze**부분에 **count** 지정
7. **Threshold method**를 선택 가능, 여기서는 **Standard Deviation** (표준편차) 선택
8. **Threshold multiplier**는 결과에 따라 수정 가능 **(e.g. too many outliers)**
   - 여기서는 2로 지정 : 2σ를 의미
9. **Sliding Window**를 선택적으로 지정할 수 있으며 필드별로 분할(그룹화) 가능
10. **Now, click on Detect Outliers**

MLTK:



MLTK:

이 실험의 **SPL**을 알고싶다면?

“Show SPL” 선택

Calculate the outliers

```
index=main source="*WinEventLog:Security" EventCode=4648 | bucket _time span=1d | stats
count by src_user _time
| eventstats avg("count") as avg stdev("count") as stdev                    // calculate the mean and standard deviation
| eval lowerBound=(avg-stdev*exact(2)), upperBound=(avg+stdev*exact(2))     // calculate the bounds as a multiple of the standard deviation
| eval isOutlier=if('count' < lowerBound OR 'count' > upperBound, 1, 0)     // mark values outside the bounds as outliers
```

MLTK:

outlier를 선택해서 outlier를 확인하기 (노란색 점 클릭)

splunk>enterprise Apps ▾ user001-splk ▾ Messages ▾ Settings ▾ Activity ▾ Help ▾ Find

Showcase Experiments Search Models Classic ▾ Settings Docs Video Tutorials Splunk Machine Learning Toolkit

New Search

Save As ▾ Create Table View Close

```
index=main source="*WinEventLog:Security" EventCode=4648
| bucket _time span=1d | stats count by src_user _time | eventstats avg("count") as avg stdev("count") as stdev  | eval lowerBound=(avg-stdev*exact(2)), upperBound=(avg+stdev*exact(2)) | eval isOutlier=if('count' < lowerBound OR 'count' >
    upperBound, 1, 0)  | search "count"="26"
```

All time ▾

✓ **185 events** (before 14/11/2023 17:36:46.000) No Event Sampling ▾ Job ▾ Smart Mode ▾

Events Patterns **Statistics (1)** Visualization

20 Per Page ▾ Format Preview ▾

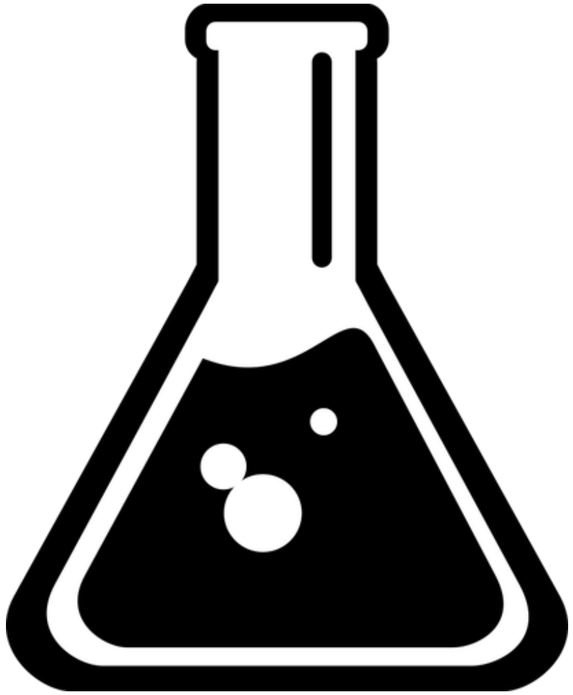
src_user	_time	count	avg	isOutlier	lowerBound	stdev	upperBound
MKRAEUSEN-D$	2020-08-27 00:00:00	26	8.80952380952381	1	-4.253692450173457	6.5316081298486335	21.872740069221077

관심있는 또 다른 필드는 무엇이 있나요?

HTTP User Agent 필드는 어떠세요?

Classification

Lab 3



Let’s experiment!

Detecting Categorical Outliers

범주형 (Categorical) 이상치 감지

MLTK:

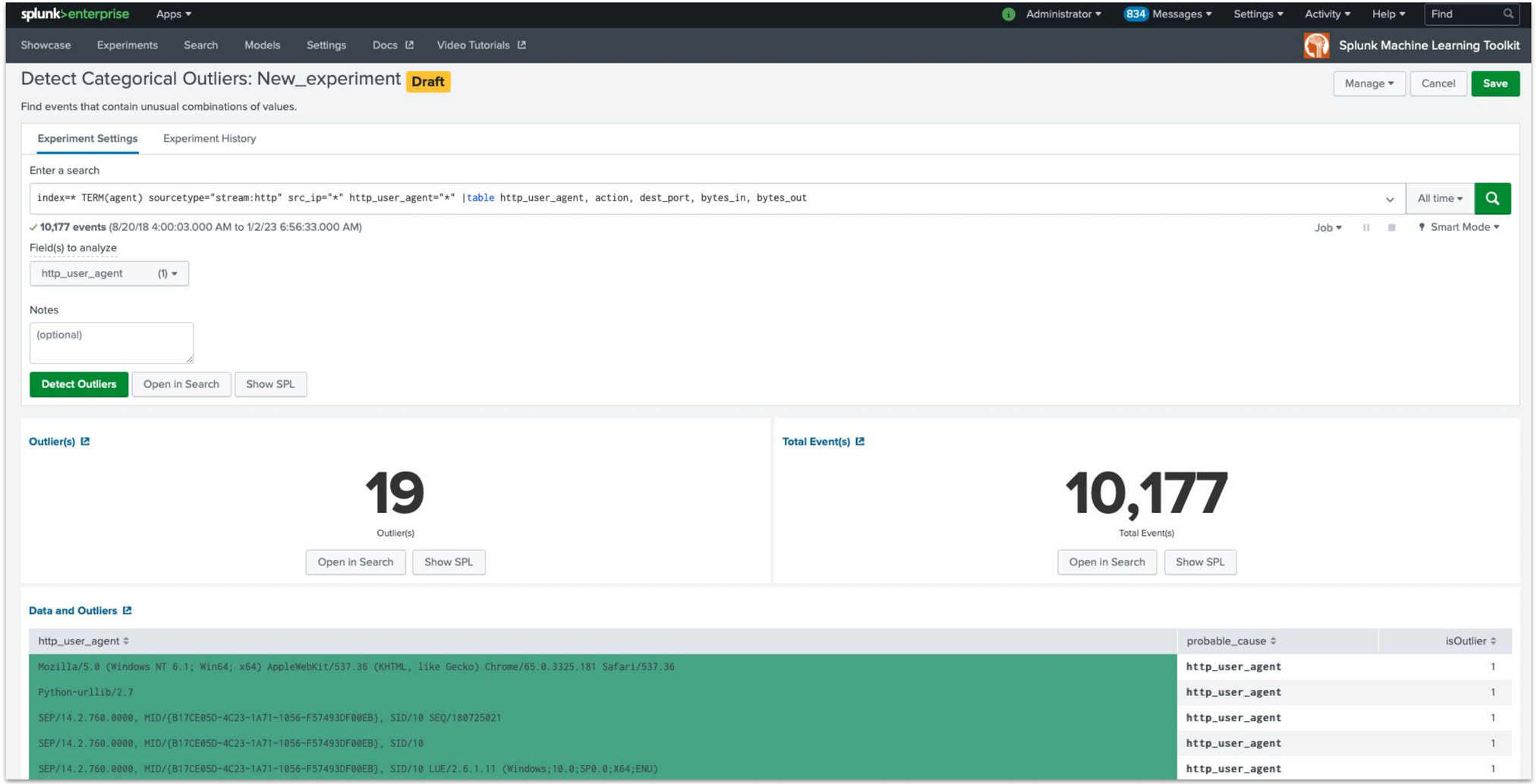
Experiment을 만들어 볼까요?

1. Experiments로 이동
2. Detect Categorical Outliers 선택
3. New Experiment 생성
4. Title 변경 (HTTP User Agent Outliers - ch)
5. 다음의 검색문 실행

```
index=* TERM(agent) sourcetype="stream:http" src_ip="*" http_user_agent="*"
| table http_user_agent, action, dest_port, bytes_in, bytes_out
```

6. Field to analyze부분에 http_user_agent 지정
7. Now, click on Detect Outliers

Detecting Categorical Outliers



Detecting Categorical Outliers

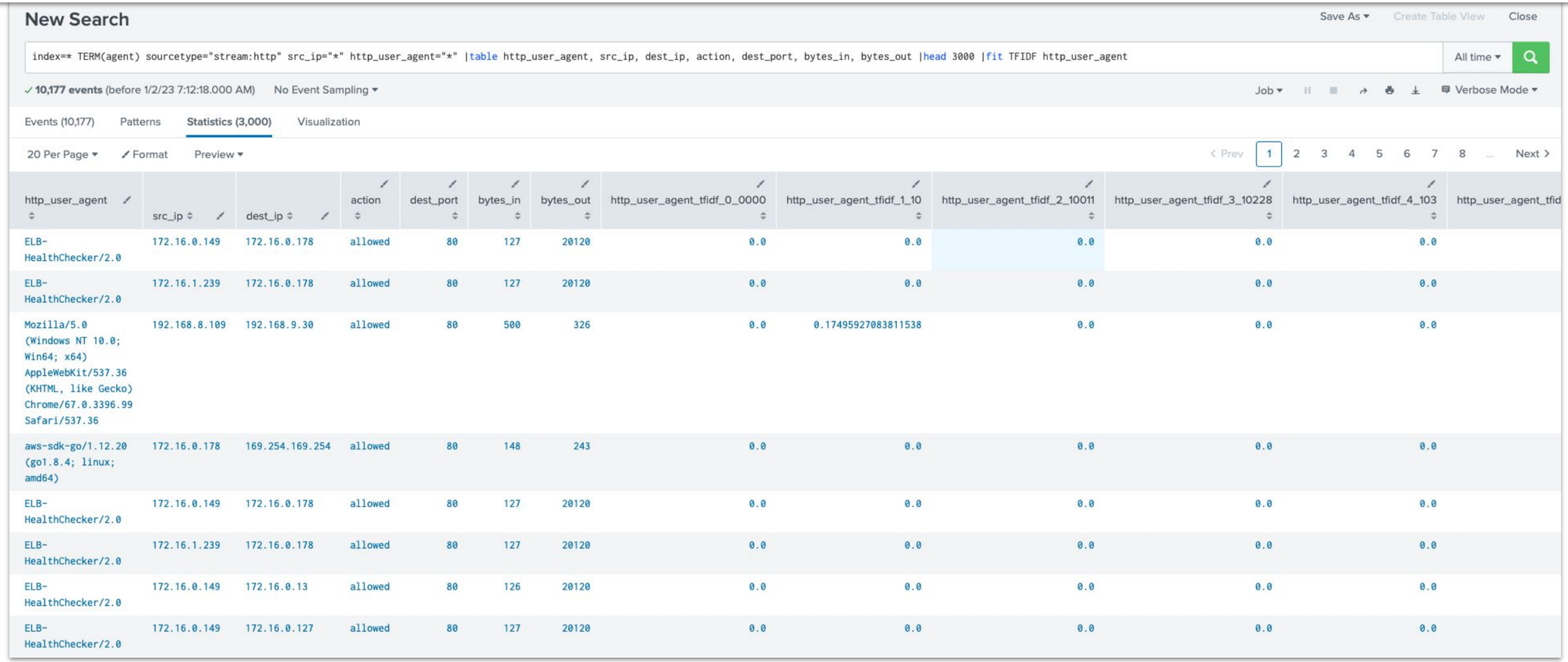
- 데이터를 검색해보고 필드를 분석해 보세요!
- 어떤 결과값이 나왔나요?
- 이 접근방식이 맞는 방식일까요?
- **Clustering (군집화)**를 통해서 접근방식을 바꿔봅시다!

Clustering

Detecting Categorical Outliers

데이터에 **TDIF** 알고리즘을 적용해 볼까요?

```
index=* TERM(agent) sourcetype="stream:http" src_ip="*" http_user_agent="*"
| table http_user_agent, src_ip, dest_ip, action, dest_port, bytes_in, bytes_out
| head 3000
| fit TFIDF http_user_agent
```



http_user_agent	src_ip	dest_ip	action	dest_port	bytes_in	bytes_out	http_user_agent_tfidf_0_0000	http_user_agent_tfidf_1_10	http_user_agent_tfidf_2_10011	http_user_agent_tfidf_3_10228	http_user_agent_tfidf_4_103	http_user_agent_tfid
ELB-HealthChecker/2.0	172.16.0.149	172.16.0.178	allowed	80	127	20120	0.0	0.0	0.0	0.0	0.0	
ELB-HealthChecker/2.0	172.16.1.239	172.16.0.178	allowed	80	127	20120	0.0	0.0	0.0	0.0	0.0	
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.99 Safari/537.36	192.168.8.109	192.168.9.30	allowed	80	500	326	0.0	0.17495927083811538	0.0	0.0	0.0	
aws-sdk-go/1.12.20 (go1.8.4; linux; amd64)	172.16.0.178	169.254.169.254	allowed	80	148	243	0.0	0.0	0.0	0.0	0.0	
ELB-HealthChecker/2.0	172.16.0.149	172.16.0.178	allowed	80	127	20120	0.0	0.0	0.0	0.0	0.0	
ELB-HealthChecker/2.0	172.16.1.239	172.16.0.178	allowed	80	127	20120	0.0	0.0	0.0	0.0	0.0	
ELB-HealthChecker/2.0	172.16.0.149	172.16.0.13	allowed	80	126	20120	0.0	0.0	0.0	0.0	0.0	
ELB-HealthChecker/2.0	172.16.0.149	172.16.0.127	allowed	80	127	20120	0.0	0.0	0.0	0.0	0.0	

Detecting Categorical Outliers

TFIDF 알고리즘이란?
https://docs.splunk.com/Documentation/MLApp/5.5.0/User/Algorithms#TFIDF:~:text=0.50%20into%20example_hard_drives_PCA_2-,TFIDF,-The%20TFIDF%20algorithm

TFIDF 알고리즘은 원시 텍스트를 숫자 필드로 변환하여 다른 머신 러닝 알고리즘과 함께 해당 데이터를 사용할 수 있도록 합니다.

TFIDF 알고리즘은 자유 형식 텍스트가 포함된 필드에서 N개의 연속 문자열(또는 용어) 그룹인 N개의 프로그램을 선택하여 머신 러닝에 적합한 숫자 필드로 변환합니다.

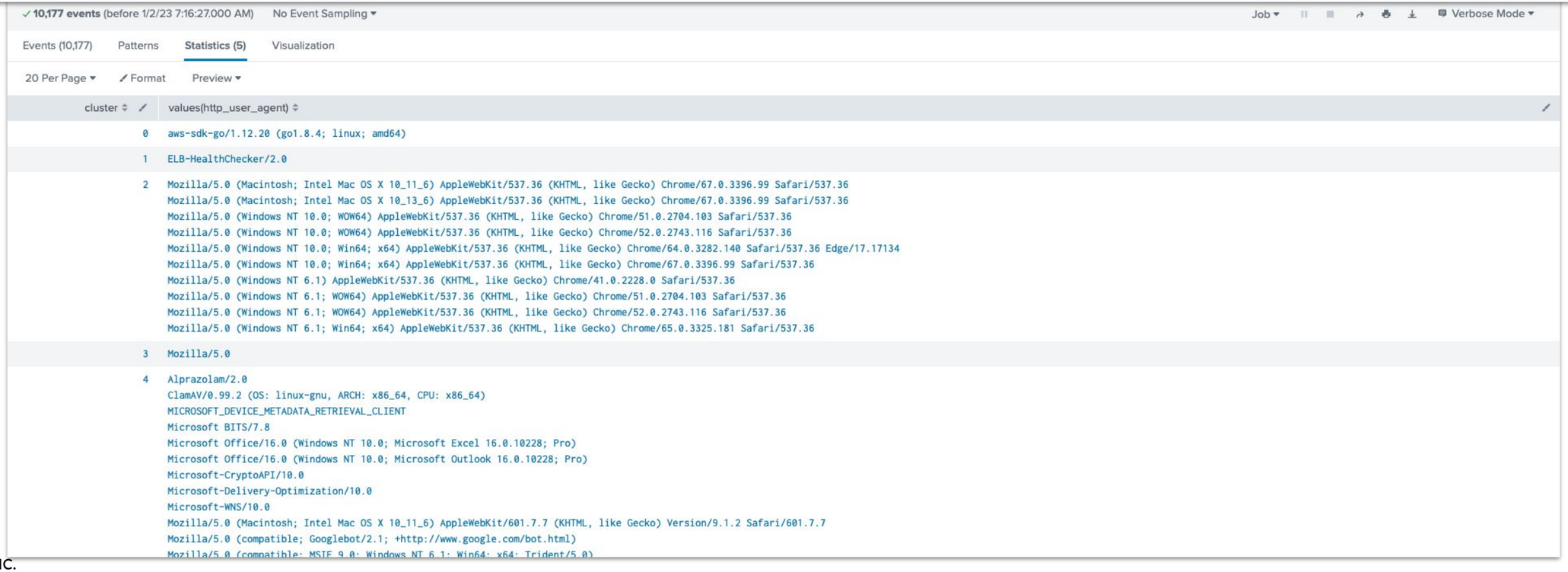
예를 들어, 이메일 피사체가 포함된 필드에서 TFIDF를 실행하면 바이그램 '프로젝트 제안'을 선택하고 각 피사체에 해당 바이그램의 가중치 빈도를 나타내는 필드를 만들 수 있습니다.

splunk에서 SPL을 통해 적용할 수 있는 머신러닝 알고리즘

Detecting Categorical Outliers

clustering(군집화)를 사용해 봅시다!

```
index=* TERM(agent) sourcetype="stream:http" src_ip="*" http_user_agent="*"
| table http_user_agent, src_ip, dest_ip, action, dest_port, bytes_in, bytes_out
| head 3000
| fit TFIDF http_user_agent
| fit KMeans k=5 http_user_agent_tfidf_*
| stats values(http_user_agent) by cluster
```



✓ **10,177 events** (before 1/2/23 7:16:27.000 AM) No Event Sampling — Job — Verbose Mode

Events (10,177) Patterns **Statistics (5)** Visualization

20 Per Page — Format — Preview

cluster	values(http_user_agent)
0	aws-sdk-go/1.12.20 (go1.8.4; linux; amd64)
1	ELB-HealthChecker/2.0
2	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.99 Safari/537.36 Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.99 Safari/537.36 Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.103 Safari/537.36 Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36 Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.140 Safari/537.36 Edge/17.17134 Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.99 Safari/537.36 Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2228.0 Safari/537.36 Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.103 Safari/537.36 Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36 Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/65.0.3325.181 Safari/537.36
3	Mozilla/5.0
4	Alprazolam/2.0 ClamAV/0.99.2 (OS: linux-gnu, ARCH: x86_64, CPU: x86_64) MICROSOFT_DEVICE_METADATA_RETRIEVAL_CLIENT Microsoft BITS/7.8 Microsoft Office/16.0 (Windows NT 10.0; Microsoft Excel 16.0.10228; Pro) Microsoft Office/16.0 (Windows NT 10.0; Microsoft Outlook 16.0.10228; Pro) Microsoft-CryptoAPI/10.0 Microsoft-Delivery-Optimization/10.0 Microsoft-WNS/10.0 Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/601.7.7 (KHTML, like Gecko) Version/9.1.2 Safari/601.7.7 Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html) Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Win64; x64; Trident/5.0)

Detecting Categorical Outliers

clustering(군집화)를 사용해 봅시다!

- KMeans 알고리즘 사용
  - 비지도 학습의 한 유형
  - 데이터를 그룹화하는 클러스터링 알고리즘으로 그룹수는 k로 표기
  - https://docs.splunk.com/Documentation/MLApp/5.5.0/User/Algorithms#TFIDF:~:text=42%20into%20cluster_model-,K%2Dmeans,-K%2Dmeans%20clustering

```
index=* TERM(agent) sourcetype="stream:http" src_ip="*" http_user_agent="*"
| table http_user_agent, src_ip, dest_ip, action, dest_port, bytes_in, bytes_out
| head 3000
| fit TFIDF http_user_agent
| fit KMeans k=5 http_user_agent_tfidf_*
| stats values(http_user_agent) by cluster
```

Detecting Categorical Outliers

Let’s try something else!

```
index=* TERM(agent) sourcetype="stream:http" src_ip="*" http_user_agent="*"
| table http_user_agent, src_ip, dest_ip, action, dest_port, bytes_in, bytes_out
| head 3000
| fit TFIDF http_user_agent
| fit KMeans k=1 http_user_agent_tfidf_*
| fields - http_user_agent_tfidf_*
| stats max(cluster_distance) by cluster http_user_agent
| sort - max(cluster_distance)
```

20 Per Page ▾ | Format | Preview ▾ | < Prev 1 2 Next >

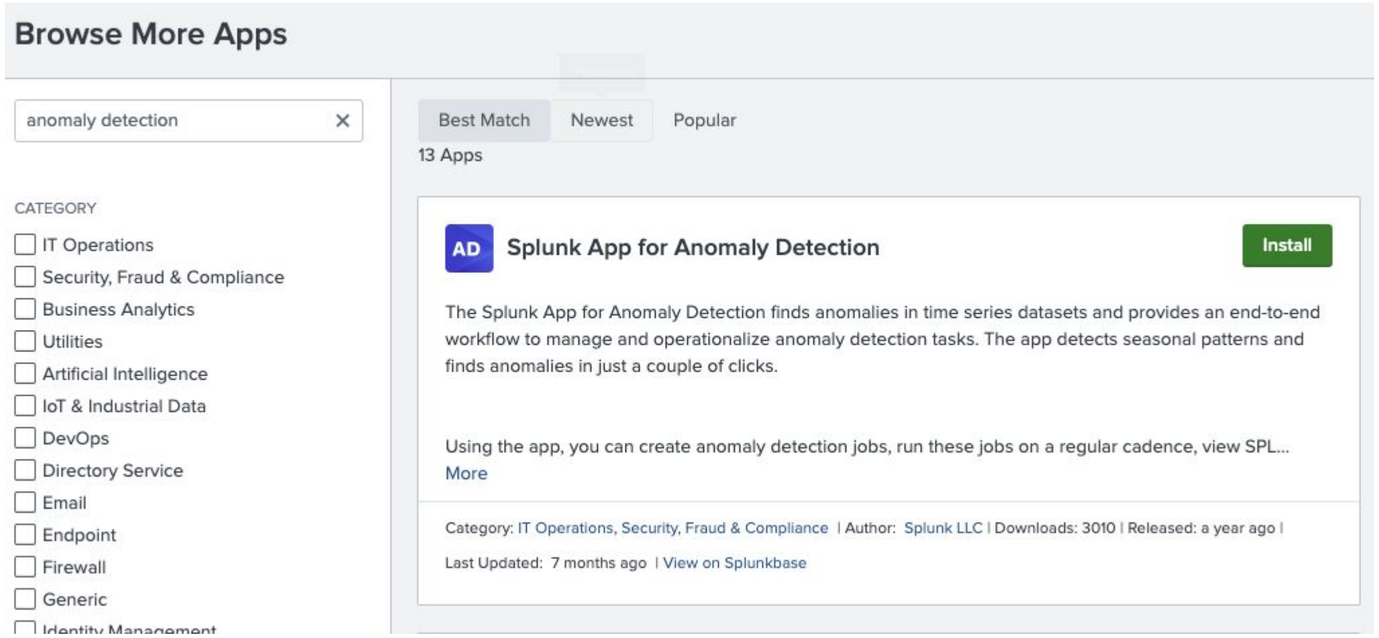
cluster	http_user_agent	max(cluster_distance)
0	Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Win64; x64; Trident/5.0)	1.1539107576147307
0	aws-sdk-go/1.12.20 (go1.8.4; linux; amd64)	1.1336852514617446
0	Mozilla/5.0	1.114913837866919
0	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.103 Safari/537.36	1.039848375036444
0	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36	1.0309951961655088
0	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.99 Safari/537.36	1.0271065926702692
0	Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.103 Safari/537.36	1.0245556136033789
0	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.99 Safari/537.36	1.0213414756302928
0	Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36	1.0154026971917771
0	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.99 Safari/537.36	0.8282086833711153
0	Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2228.0 Safari/537.36	0.8158749659787358
0	Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/65.0.3325.181 Safari/537.36	0.790346971694435
0	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.140 Safari/537.36 Edge/17.17134	0.7191211131029
0	ELB-HealthChecker/2.0	0.3957707127071142
0	Alprazolam/2.0	0.31577071270711504

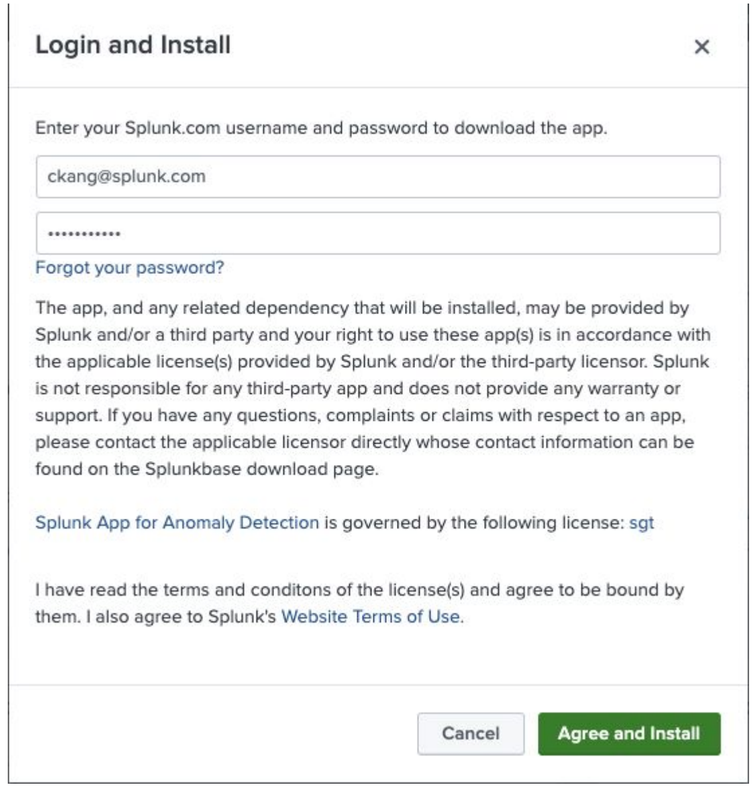
Anomaly Detection

Splunk App for Anomaly Detection

설치하기

1. Apps > Browse more apps > `**anomaly detection**` 검색 > Install클릭 > login에 Splunk 가입시 사용한 username/password 기입 > Open app

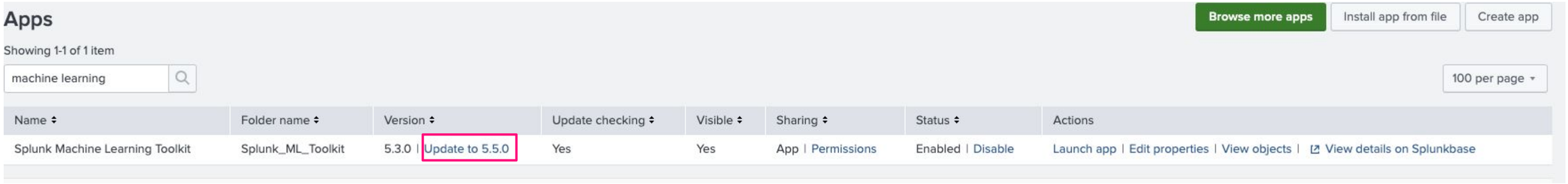


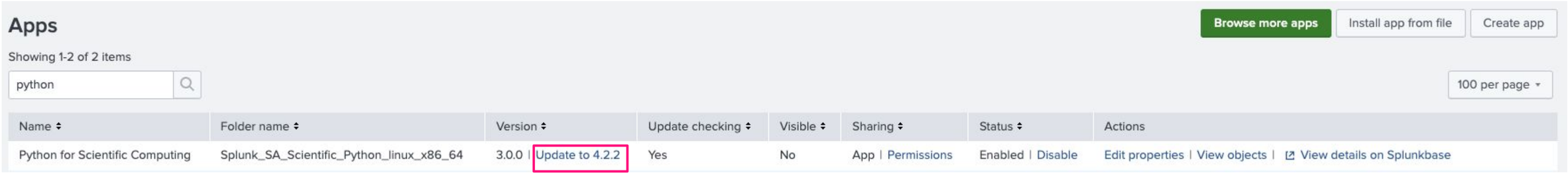


Splunk App for Anomaly Detection

관련 앱 Update

- Splunk Machine Learning Toolkit과 Python for Scientific Computing update
  1. Apps > Splunk Machine Learning Toolkit > Edit properties > Update checking을 yes로 변경
  2. Apps > Splunk Machine Learning Toolkit > Edit properties > Update checking을 yes로 변경
  3. Splunk > Settings > Server controls 에서 splunk server restart
  4. 위의 두 앱을 검색하여 **Update** 진행





Splunk App for Anomaly Detection

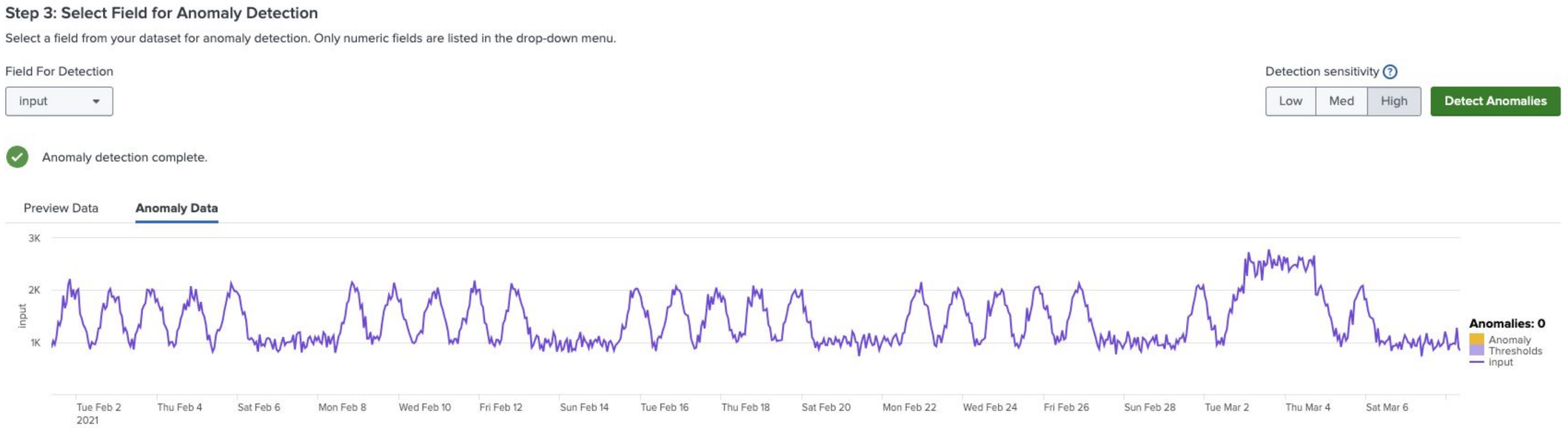
Step 0: Splunk App for Anomaly Detection으로 이동

Step 1: Create a New Job > Job Name: test_[user 명]

Step 2: Add the Dataset (시간대는 All time으로 변경)

```
| inputlookup kpi.csv
```

Step 3: Field For Detection를 input으로 지정하고 Detect Anomalies



Splunk App for Anomaly Detection

Password_Spraying_Attacks 으로 응용

Step 0: Splunk App for Anomaly Detection으로 이동

Step 1: Create a New Job > Job Name: Password_Spraying_Attacks_[user 명]

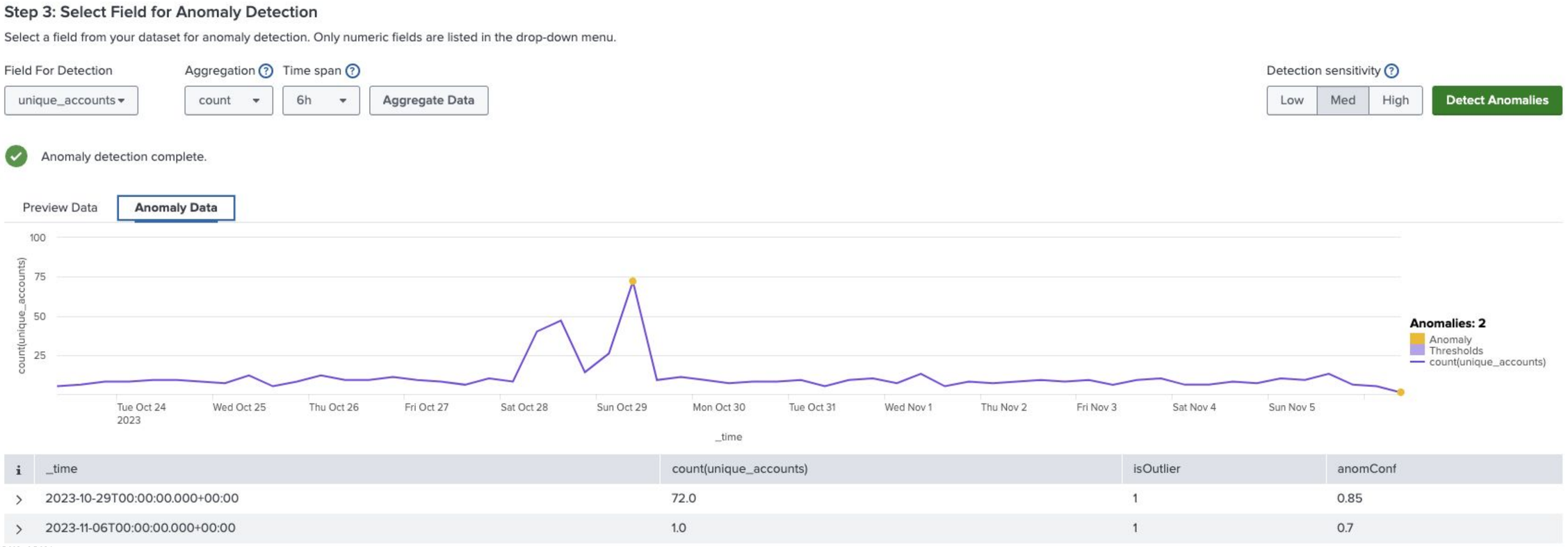
Step 2: Add the Dataset (시간대는 All time으로 변경)

```
index="main" sourcetype="XmlWinEventLog_ws" EventCode=4625 LogonType=3
| bucket span=2m _time
| stats dc(TargetUserName) as unique_accounts values(TargetUserName) as
  tried_accounts by _time, IpAddress, LogonType, dvc
```

Splunk App for Anomaly Detection

Step 3: Select Field for Anomaly Detection 에 아래와 같이 설정 후 Detect Anomalies

- Field For Detection: unique_accounts
- Aggregation: count
- Time span: 6h



i	_time	count(unique_accounts)	isOutlier	anomConf
>	2023-10-29T00:00:00.000+00:00	72.0	1	0.85
>	2023-11-06T00:00:00.000+00:00	1.0	1	0.7

Anomaly Detection

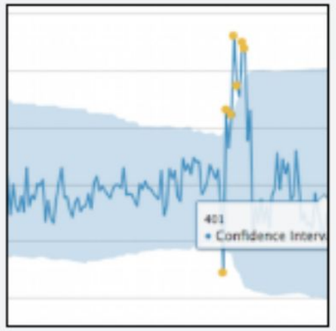
Numeric Outlier Detection

이상 징후 탐지의 한 유형

일부 숫자 값의 이상값

- 트랜잭션 수
- 트랜잭션 지연 시간
- 시스템 사용률(CPU/메모리)
- 로그인 횟수
- 데이터 전송량
- 작업 간 시간
- 센서 측정

MLTK의 숫자 이상값 탐지 지원



Detect Numeric Outliers

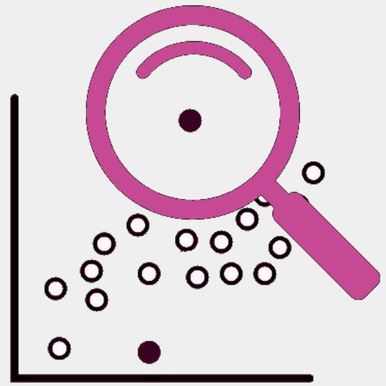
Find values that differ significantly from previous values.

Examples

- Detect Outliers in Server Response Time
- Detect Outliers in Number of Logins (vs. Predicted Value)
- Detect Outliers in Supermarket Purchases
- Detect Outliers in Power Plant Humidity
- Detect Cyclical Outliers in Call Center Data
- Detect Cyclical Outliers in Logins

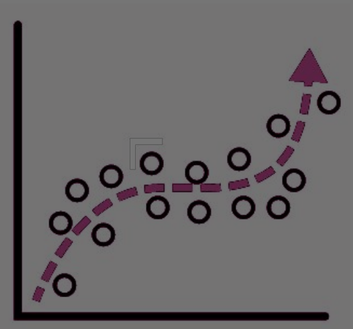
Anomaly Detection

Anomaly detection - 이상 탐지



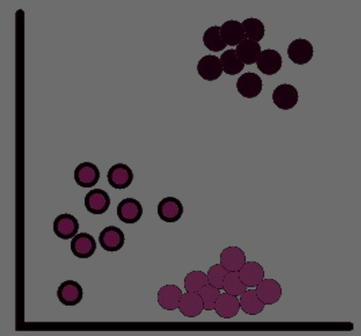
- 과거 행동으로부터의 이탈
- 동료들로부터의 이탈 (Multivariate AD 또는 Cohesive AD로도 알려짐)
- 기능에서의 비정상적인 변화

Predictive Analytics - 예측 분석



- 서비스 상태 점수/이탈 예측
- 이벤트 예측
- 트렌드 예측
- 영향을 끼치는 엔티티 탐지
- 고장 조기 경고

Clustering - 군집화



- 동료 그룹 식별
- 이벤트 상관관계
- 경보 감소
- 행동 분석

Outlier vs Anomaly

뭐가 다를까요?

Outlier (noun):

- 샘플 내 다른 값들과 **뚜렷하게 다른 값을 가진** 통계적 관찰값
- 이상치는 일반적으로 단일 값이나 측정값에서 예상치 못한 것을 의미합니다

Anomaly (noun):

- 다르거나, 비정상적이거나, 특이하거나, **쉽게 분류되지 않는 것**
- 일반적인 규칙에서의 벗어남
- 이상은 보통 여러 관찰값들의 집합을 기반으로 합니다
-

모든 **Outlier**가 **anomaly**는 아니지만 **anomaly**는 종종 **outlier**로 구성되거나 포함합니다.

Cheat sheet for anomaly detection in Splunk

Command	Description
analyzefields (af)	Analyze numerical fields for their ability to predict another discrete field.
anomalies	Computes an "unexpectedness" score for an event.
anomalousvalue	Finds and summarizes irregular, or uncommon, search results.
anomalydetection	Identifies anomalous events by computing a probability for each event and then detecting unusually small probabilities.
cluster	Clusters similar events together.
kmeans	Performs k-means clustering on selected fields.
outlier	Removes outlying numerical values.
rare	Displays the least common values of a field.

https://docs.splunk.com/Documentation/Splunk/latest/SearchReference/Commandsbycategory

Method / Algorithm	Description
DensityFunction	The DensityFunction algorithm provides a consistent and streamlined workflow to create and store density functions and utilize them for anomaly detection…
LocalOutlierFactor	The LocalOutlierFactor algorithm measures the local deviation of density of a given sample with respect to its neighbors…
OneClassSVM	The OneClassSVM algorithm fits a model from a set of features or fields for detecting anomalies and outliers…
Clustering Algorithms	Spot point anomalies or anomaleous clusters. Inspect e.g. cluster_distance with KMeans, cluster=-1 with DBSCAN…
Classifiers and Regressors	Inspect strong residuals when applying your well fitted model to new incoming data points.
ML SPL API	Wrap your own algorithms of choice

https://docs.splunk.com/Documentation/MLApp/latest/User/Algorithms

Enterprise Security로 이상치 탐지

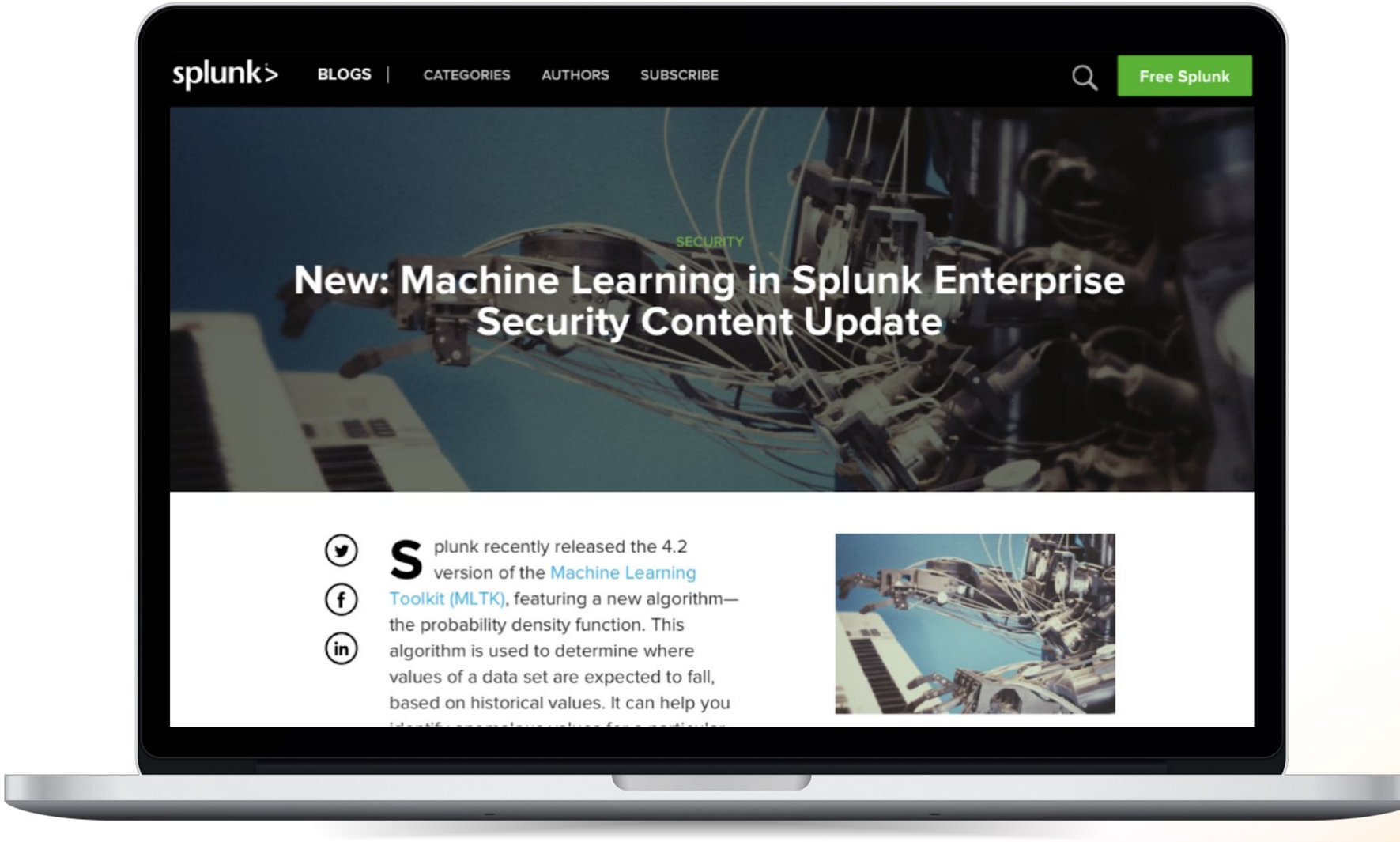
Enterprise Security 버전 6.0부터 머신 러닝 툴킷(MLTK)과 함께 제공되는 기능

Splunk 머신 러닝 툴킷(MLTK)이 ES(Enterprise Security)의 모델 생성 패키지로 익스트림 검색(XS)을 대체합니다. MLTK는 더 큰 규모로 확장할 수 있으며 모델을 통해 더 많은 비정상 이벤트를 식별할 수 있습니다.

ES는 내부적으로 MLTK DensityFunction을 사용하고 있습니다.

https://docs.splunk.com/Documentation/ES/latest/Admin/MLTKoverview

ES Content Update includes Density Function content



Splunkbase에서 머신 러닝과 관련된 앱들

splunkbase™

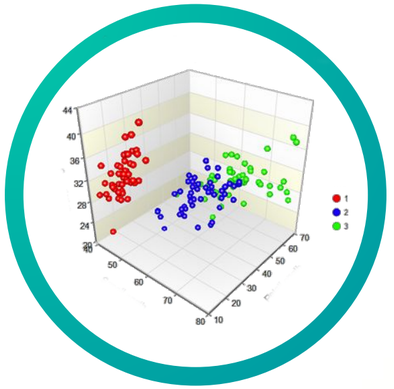
Machine Learning Applications

Splunk Security Essentials

DGA App for Splunk

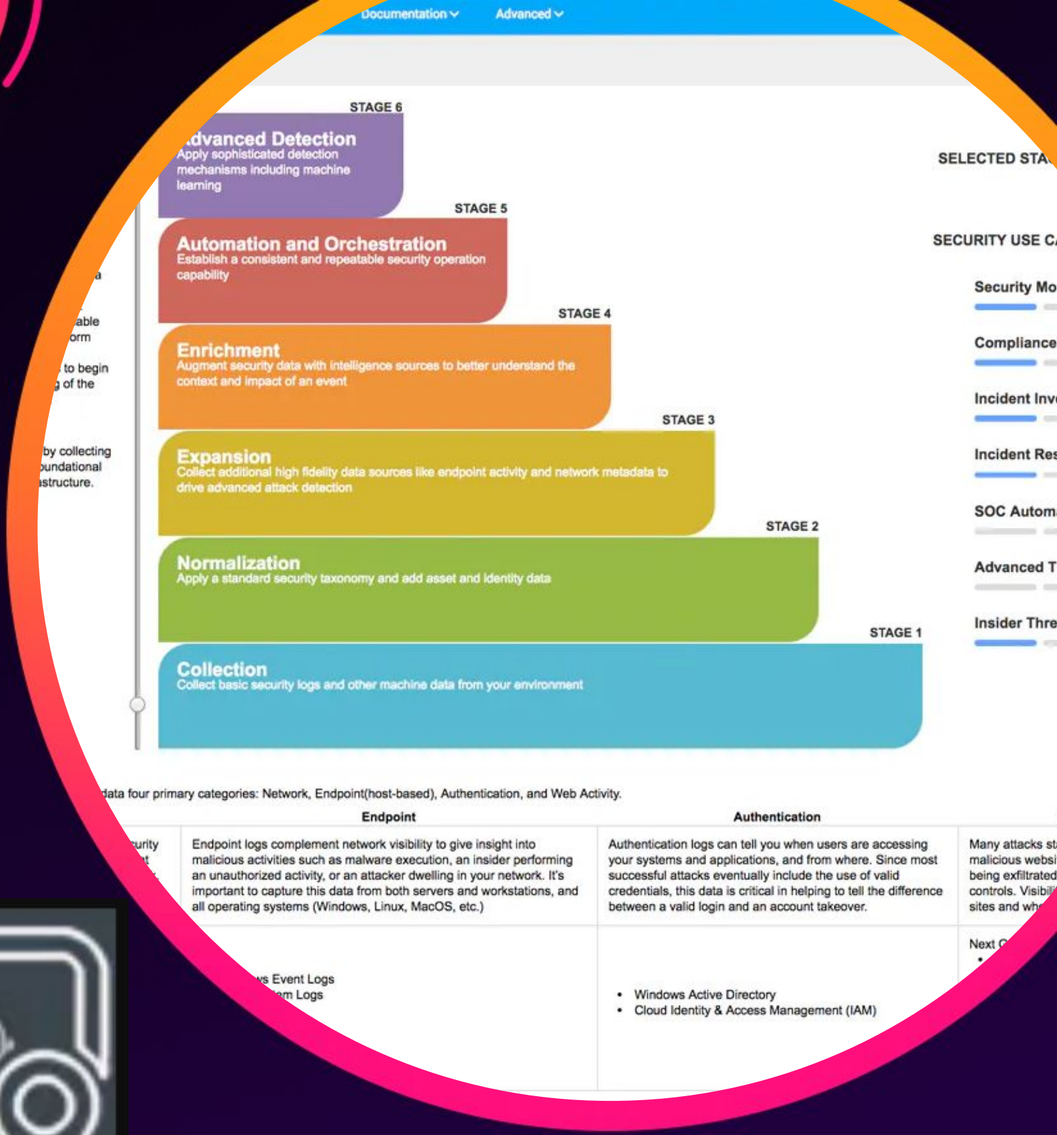
Botnet App for Splunk

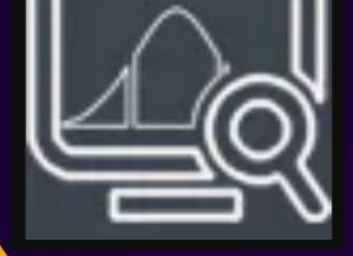
Custom Detections using MLTK



Splunk Security Essentials

Download from Splunkbase:
https://splunkbase.splunk.com/app/3435

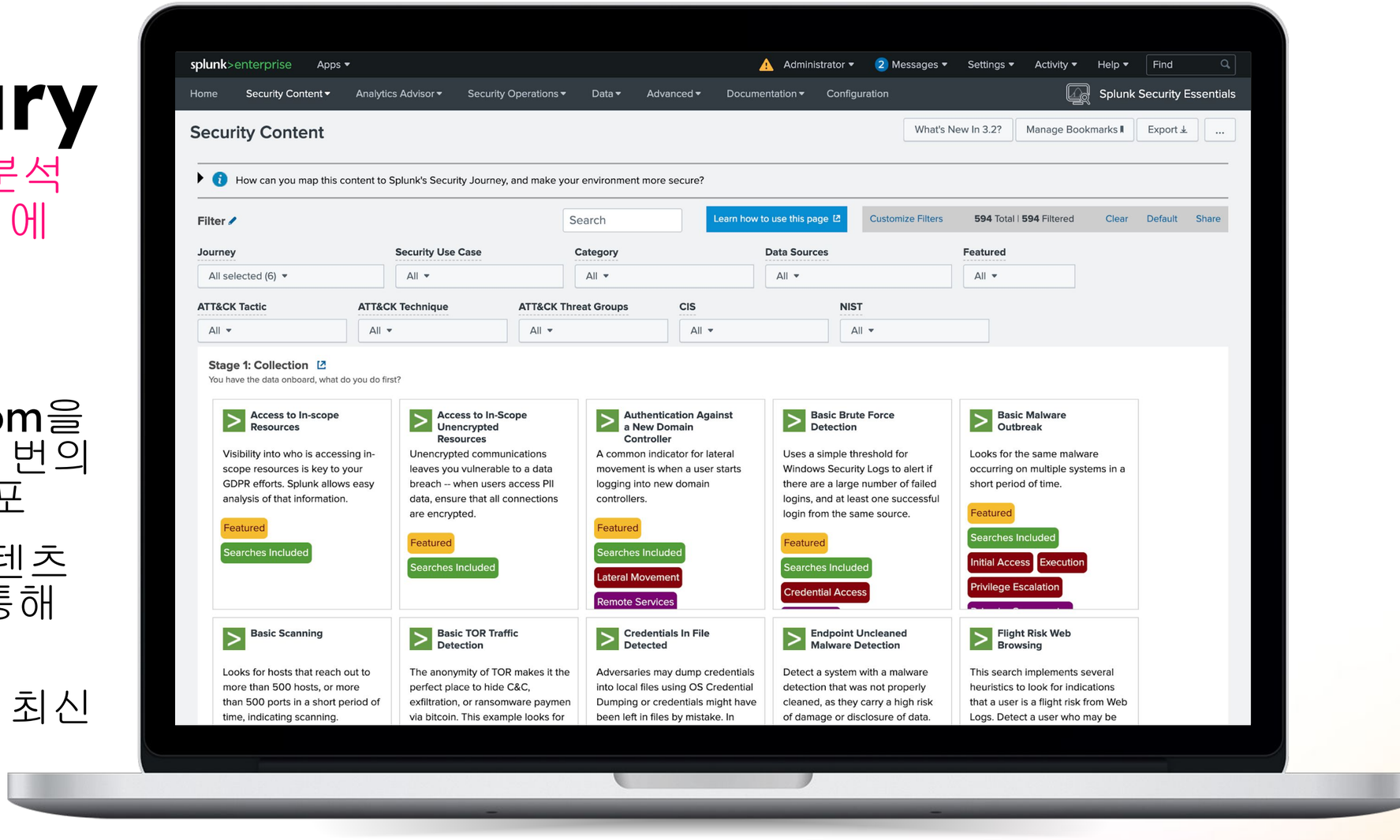




Security Content Library

900개 이상의 보안 탐지 및 분석 스토리를 찾아보고, 즐겨찾기에 추가하고, 배포하세요.

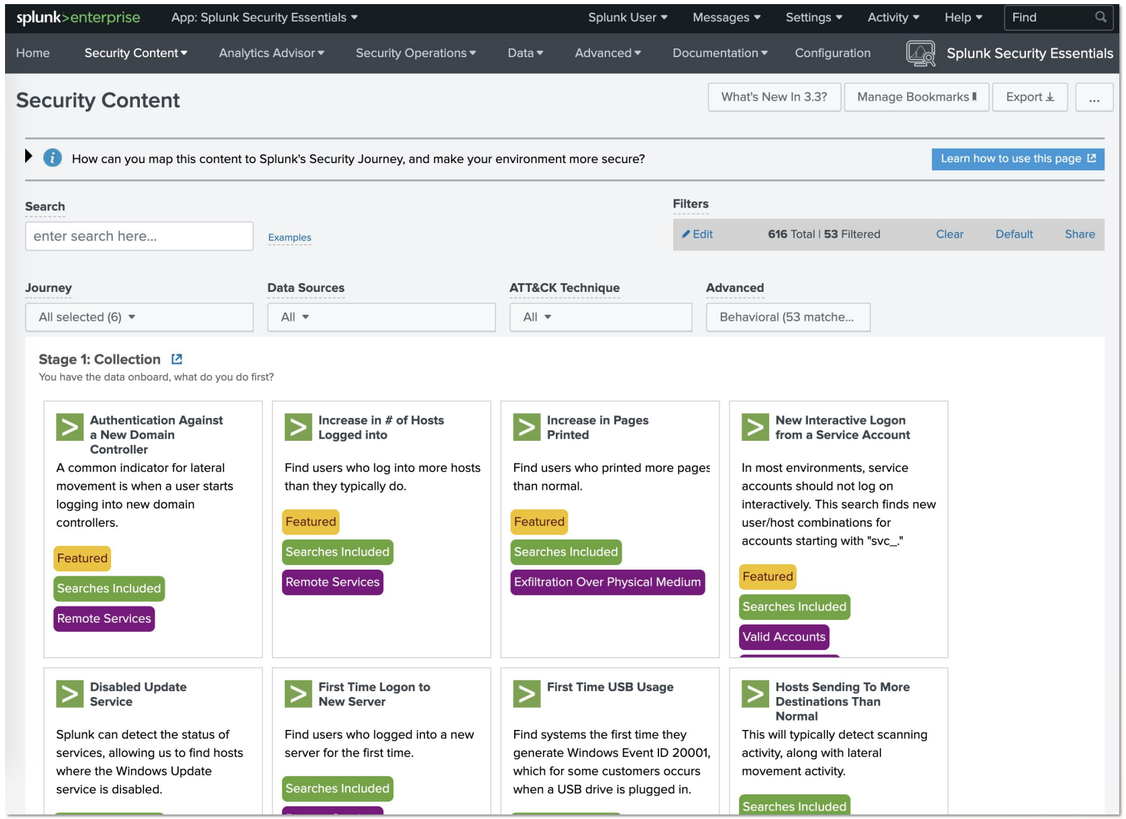
- Splunk Cloud, Enterprise Security, UEBA 및 Phantom을 위한 보안 콘텐츠 저장소 몇 번의 클릭만으로 보안 콘텐츠 배포
- 중요 이벤트를 강화하고 콘텐츠 라이브러리의 컨텍스트를 통해 분석 실행
- 기존 및 새로운 위협에 대한 최신 정보를 유지하세요.



Includes Behavioral Detections

환경에서 악의적인 사용자를 식별하세요:

- **UEBA** 제품에서 공통적으로 사용되는 **50**개 이상의 사용 사례를 포함하며, 모두 **Splunk Enterprise**를 사용합니다.
- 외부 공격자 및 내부자 위협 대상
- 소규모 기업부터 대규모 기업까지 확장 가능
- 앱에서 저장, **ES/UBA**로 결과 전송

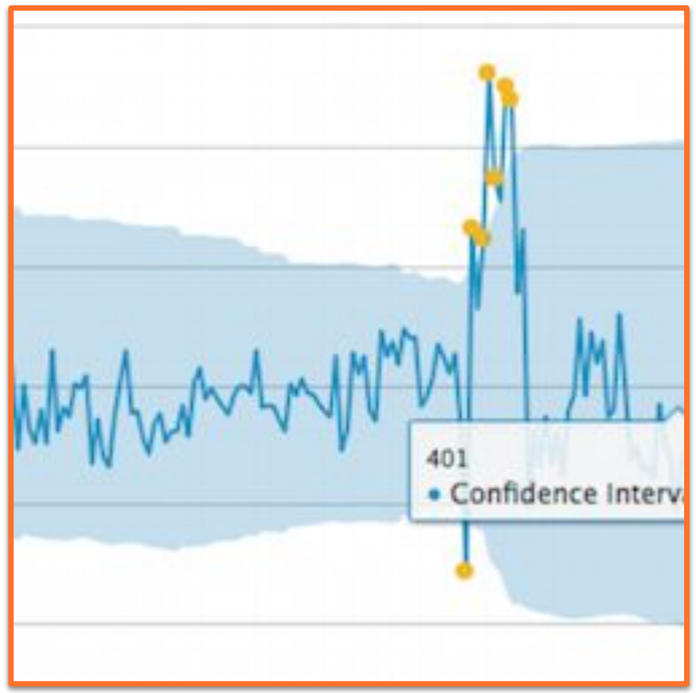


지금 바로 사용 사례를 해결한 다음 고급 ML 탐색을 위한 Splunk UBA를 사용하세요.

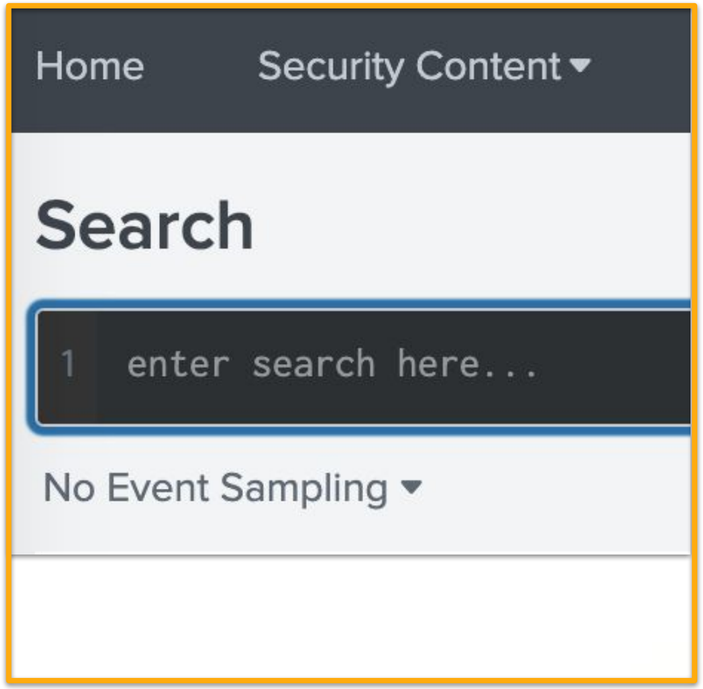
Splunk Security Essentials

Types of Use Cases

통계로 처음 보기



표준 편차를 사용한
시계열 분석

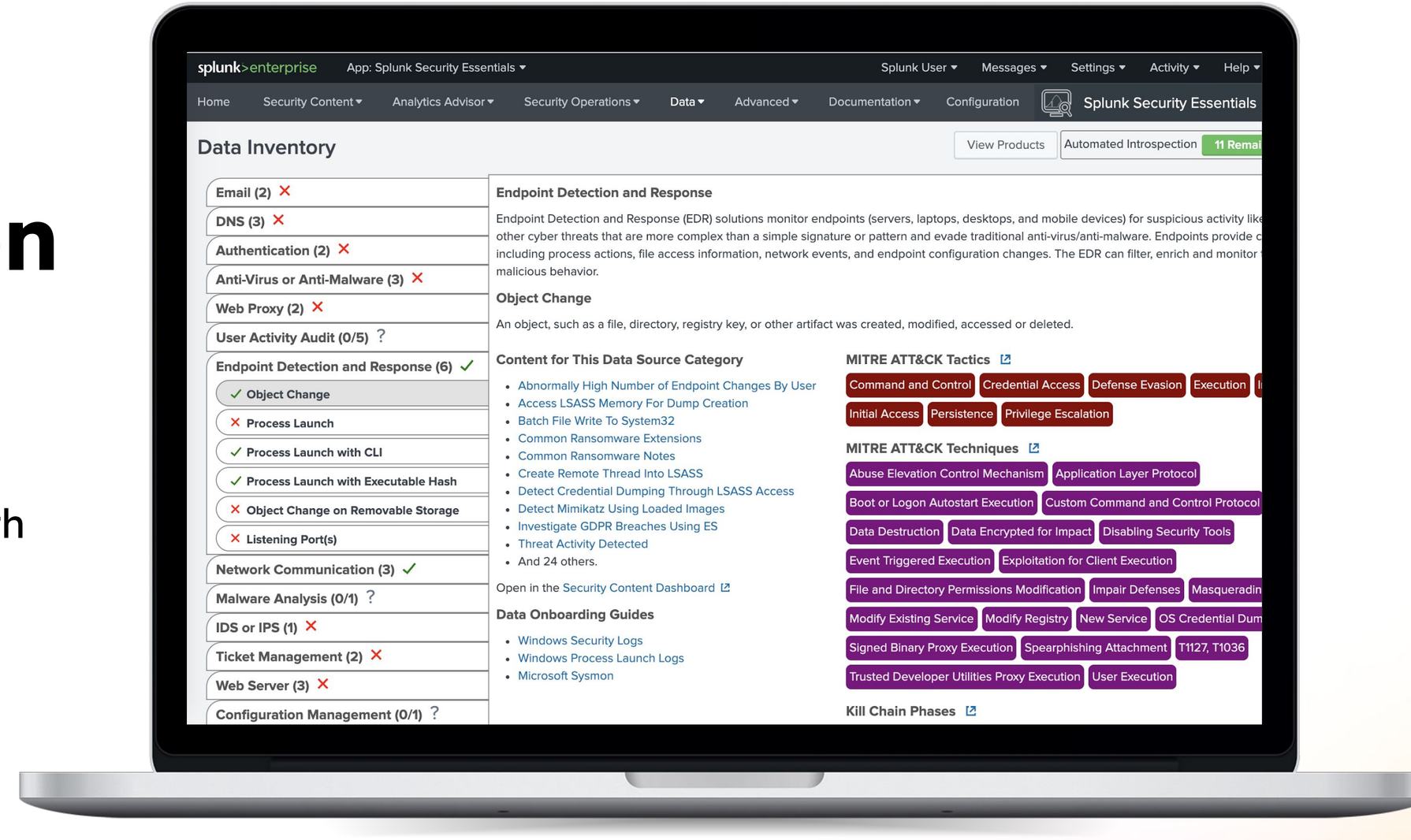


일반 보안 분석 검색

Data and Content Introspection

데이터와 저장된 검색을 추적하여 가시성을 확보하세요.

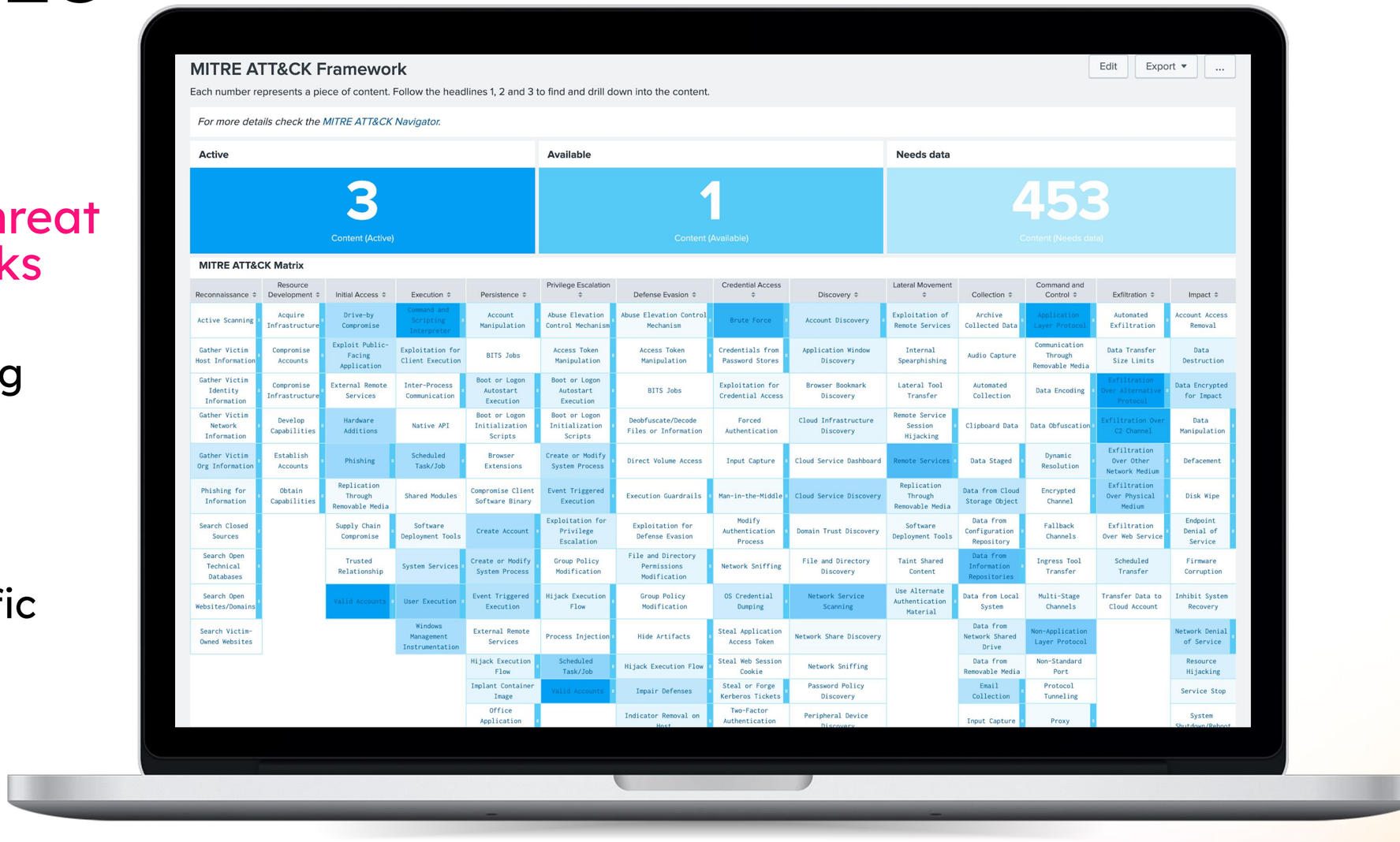
- Enrich saved searches with tags and metadata
- Automatic categorization by security products
- Scale with data model acceleration and by reinforcing CIM compliance



Operationalize Security Frameworks

Identify gaps, improve threat detection, and reduce risks

- Develop an understanding of your security posture against MITRE ATT&CK® and Cyber Kill Chain® frameworks
- Find detections for specific Threat Groups or Threat Software
- Drilldown on known Tactics and Techniques and Kill Chain Phases for more details



DGA App for Splunk

Download from Splunkbase:
https://splunkbase.splunk.com/app/3559/

Domain Generating Algorithms (DGA)

What's the problem?

DGA를 탐지하기 위한 도전 과제:

- 잠재적으로 무한한 블랙리스트 항목에 대한 정적 매칭 실행
- 정규식을 사용하면 이 목록을 좁힐 수 있지만 여전히 규칙을 계산하고 찾기가 어렵습니다 (규칙에 대한 예외를 정의하기도 어렵습니다).
- 알 수 없는 미지수가 있나요?
- 모호한게 있나요?
- ML을 해보세요!

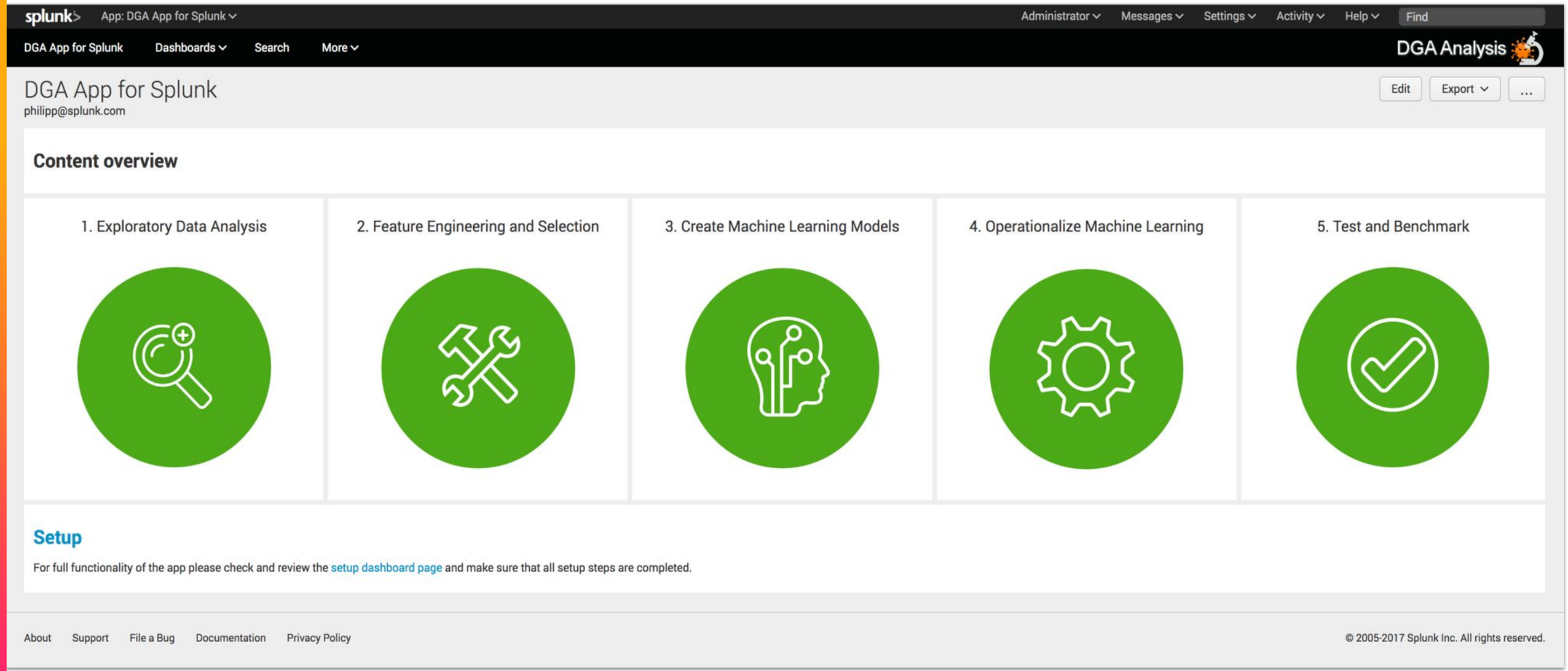
Example of DGAs:

domain
iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea
ifferfsodp9ifjaposdfjhgosurijfaewrwergwea
ayylmaotjhsstasdfasdfasdfasdfasdfasdfasdf
lazarusse.suiche.sdfjhgosurijfaqwqwqrgwea
sdfjhgosurijfaqwqwqrgwea

Example IoCs for Wannacry: https://cert.europa.eu/static/SecurityAdvisories/2017/CERT-EU-SA2017-012.pdf

Example IoCs for Sunburst: https://github.com/fireeye/sunburst_countermeasures/blob/main/indicator_release/Indicator_Release_NBIs.csv

Additional details at https://www.splunk.com/en_us/blog/security/sunburst-backdoor-detections-in-splunk.html



- Example for end to end data science process
- Disclaimer: this is not a turn key solution but a template to get you started
- Feel free to improve and give us your feedback!

splunk> App: DGA App for Splunk | Administrator | Messages | Settings | Activity | Help | Find

DGA App for Splunk | Dashboards | Search | More | DGA Analysis

1. Data Exploration

Edit | Export | ...

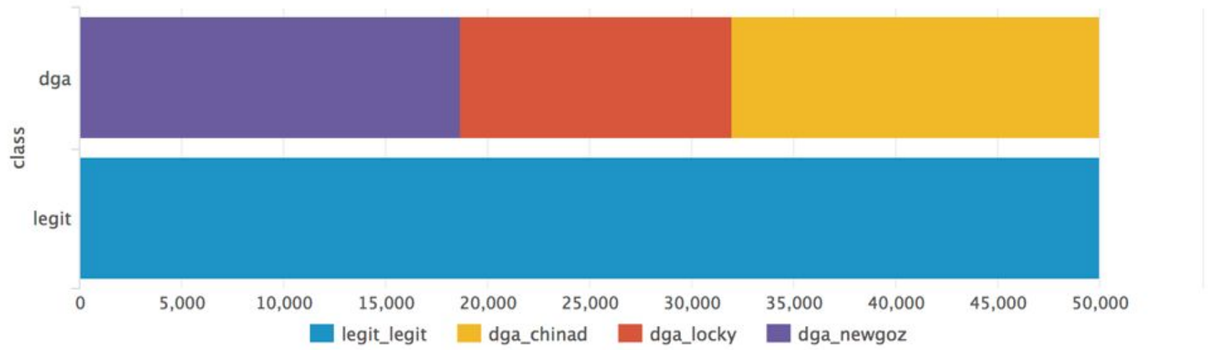
Dataset Overview

The dataset consists of a labeled domain names that indicate whether a domain is legit or created by some DGA that is known from botnets. We have around 60% domain names from legit domains and remaining 40% split across 3 DGA subclasses that correspond to different botnets.

domains.csv

class	domain	subclass
legit	google.com	legit
legit	www.google.com	legit
legit	microsoft.com	legit
legit	facebook.com	legit
legit	doubleclick.net	legit

« prev 1 2 3 4 5 6 7 8 9 10 next »



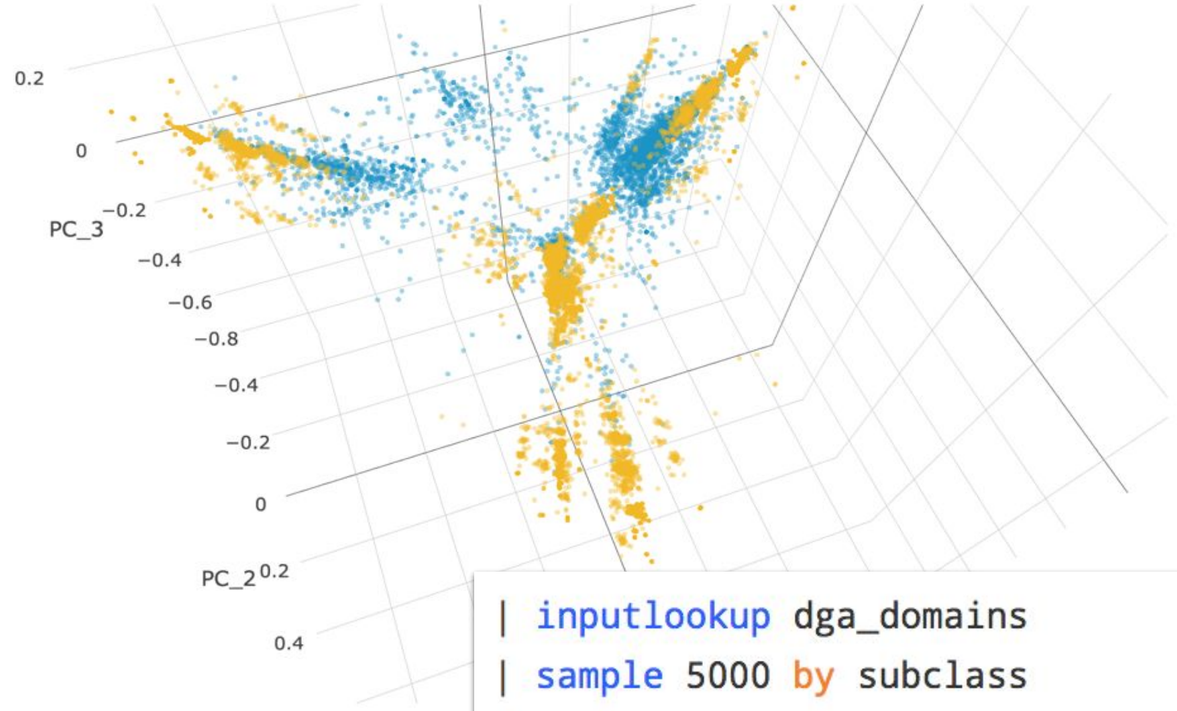
Sample of domains by subclasses

column	chinad	legit	locky	newgoz
domains	15nrbrfoqajvt92p.com	943.engine.mobileapptracking.com	clgsaguvihthkai.click	15kfdx1uy33wsq420w81t24jmz.biz
	2eb7jd4214efdemz.net	a.applvn.com	eltineojriud.biz	1c4pu7etfir7l106r8vqti89xz.net
	9dkwgr7osoqr3sj8.info	ads.rubiconproject.com	jpyeecccd.biz	1i14a381pwjcnd1pyddk2191bcng.biz
	bof8b9ediq2zgxcx.info	buy.tinypass.com	jtxopqwxokgcdnqmn.pw	1jjxlm24webiq1m91b7v1u9i37h.org
	eejxv00hk3e9mu7n.net	fmcz4-1.fna.fbcdn.net	lmhyhqym.su	1wyjzq3104oa5hn7rpkw1w36niz.com
	eo9i0xeybpezdwjx.org	homedepot.az1.qualtrics.com	okfhcdiayd.click	mokou21vftme61whl7hggd7wp0.net
	jd59a3g1bqumyvnl.org	pandora.com	pchjmkjhoqt.biz	p2ht3uqv42qx14vveb5nwlasr.org
	kdh2kxsxrohdr432.net	pix.bfi0.com	tvmlivvyb.biz	q21qhn15f53lz17y86983vc31e.com
	s45srq078fv7q7j8.net	track.wattpad.com	vawmxlhoiconu.org	ustqjsa8rvla1y25kr386qngm.com
	swfa7qz5k2pefj63.biz	us-east-1.elb.amazonaws.com	xxixgjtltu.pl	zrxndhyj6yb3199ycn1xve3m3.org

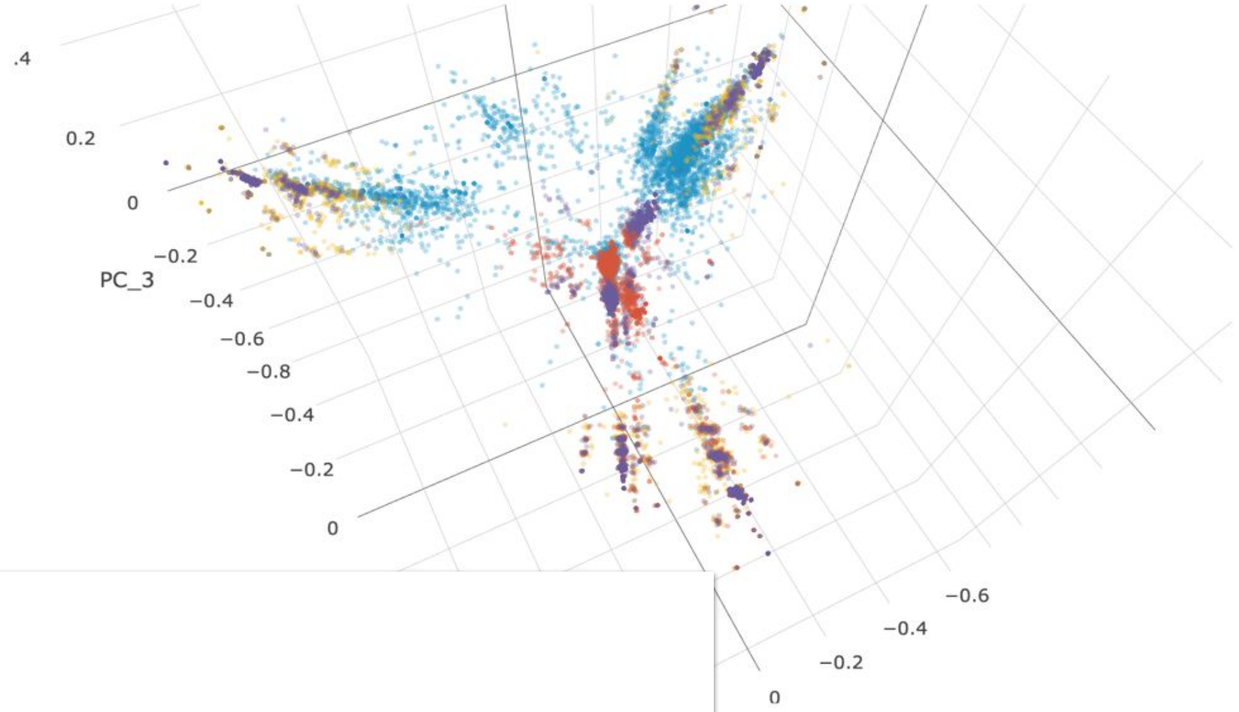
- Small data set with 100K domain names for training and testing

Data Exploration with Text Mining approach

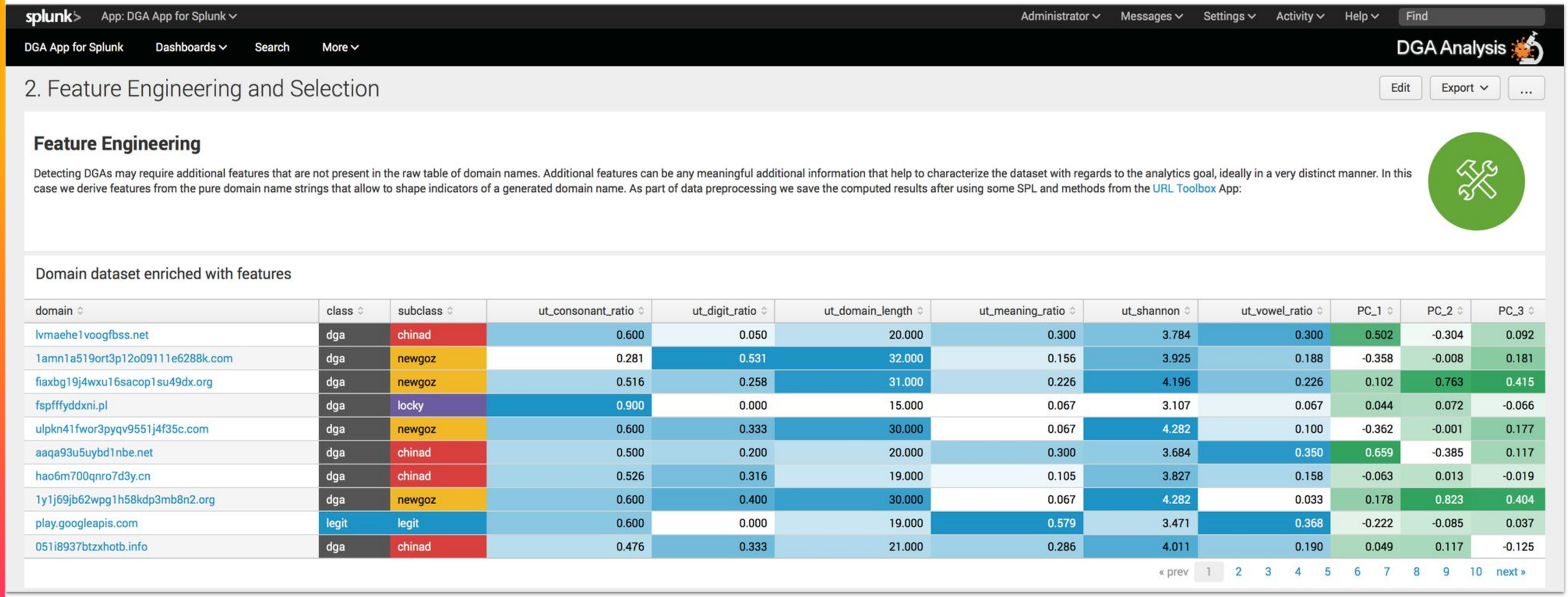
n-gram analysis (2-3 char groups) of domain names with PCA k=3 by class



n-gram analysis (2-3 char groups) of domain names with PCA k=3 by subclass




```
| inputlookup dga_domains
| sample 5000 by subclass
| fit TFIDF domain analyzer=char ngram_range=2-3 into "dga_ngram"
| fit PCA domain_tfidf* k=3 into "dga_pca"
| fields - domain_tfidf*
| eval label=class."_".subclass
| table label PC_*
| sort 0 - label
```

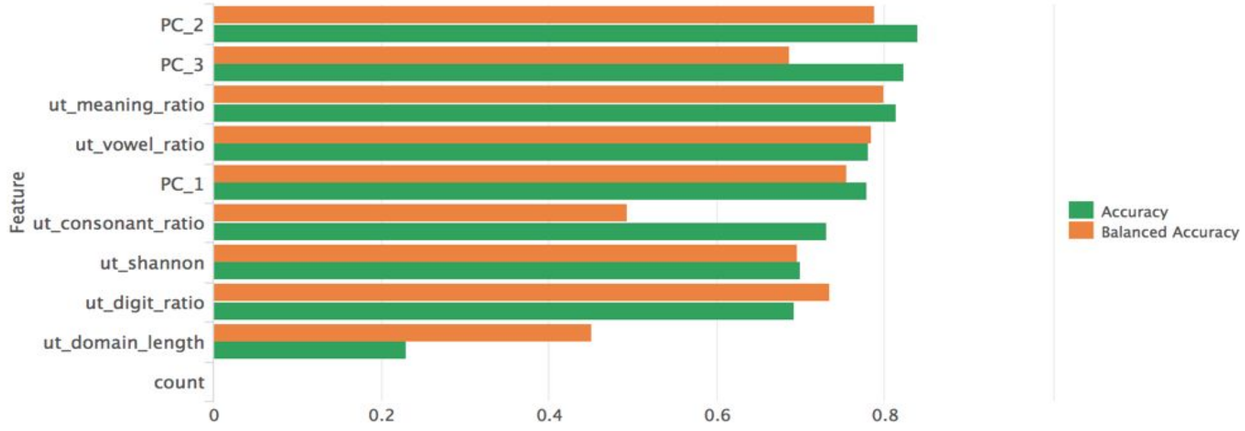


domain	class	subclass	ut_consonant_ratio	ut_digit_ratio	ut_domain_length	ut_meaning_ratio	ut_shannon	ut_vowel_ratio	PC_1	PC_2	PC_3
lvmaehe1voogfbss.net	dga	chinad	0.600	0.050	20.000	0.300	3.784	0.300	0.502	-0.304	0.092
1amn1a519ort3p12o09111e6288k.com	dga	newgoz	0.281	0.531	32.000	0.156	3.925	0.188	-0.358	-0.008	0.181
fiaxbg19j4wxu16sacop1su49dx.org	dga	newgoz	0.516	0.258	31.000	0.226	4.196	0.226	0.102	0.763	0.415
fspfffyddxni.pl	dga	locky	0.900	0.000	15.000	0.067	3.107	0.067	0.044	0.072	-0.066
ulpkn41fwor3pyqv9551j4f35c.com	dga	newgoz	0.600	0.333	30.000	0.067	4.282	0.100	-0.362	-0.001	0.177
aaqa93u5uybd1nbe.net	dga	chinad	0.500	0.200	20.000	0.300	3.684	0.350	0.659	-0.385	0.117
hao6m700qnro7d3y.cn	dga	chinad	0.526	0.316	19.000	0.105	3.827	0.158	-0.063	0.013	-0.019
1y1j69jb62wpg1h58kdp3mb8n2.org	dga	newgoz	0.600	0.400	30.000	0.067	4.282	0.033	0.178	0.823	0.404
play.googleapis.com	legit	legit	0.600	0.000	19.000	0.579	3.471	0.368	-0.222	-0.085	0.037
051i8937btzxhotb.info	dga	chinad	0.476	0.333	21.000	0.286	4.011	0.190	0.049	0.117	-0.125

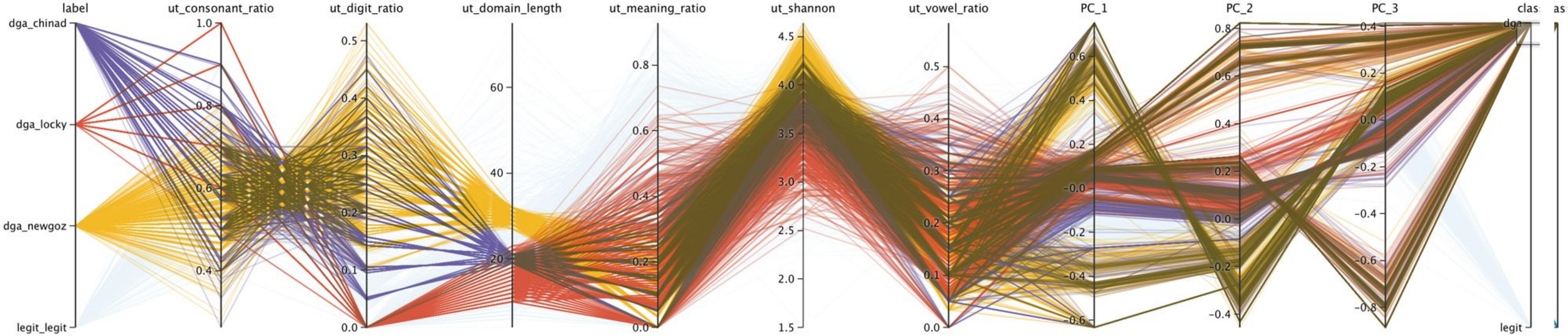
- More features can significantly improve your machine learning models
- Extend this with your feature engineering ideas (e.g. subdomains, age of domain registration, rating/scoring from threatlists for known malicious domains etc.)

Distribution of classes depending on example feature combination

Identify useful features for classification with the analyzefields command



Parallel coordinate chart of classes and top features



Currently showing 3000 / 4000 datapoints Clear filters

3. Create Machine Learning Models

Edit Export ...

Training and evaluation of different machine learning models

We train 4 machine learning models on the same data set using different algorithms for classification. Using a 50:50 split we can evaluate which models perform better and have a lower error rate.

Train models

1. Random Forest Classifier

class	predicted(class)=dga	predicted(class)=legit
dga	24693	226
legit	392	24681

2. Support Vector Machine

class	predicted(class)=dga	predicted(class)=legit
dga	23988	931
legit	1171	23902

3. Decision Tree Classifier

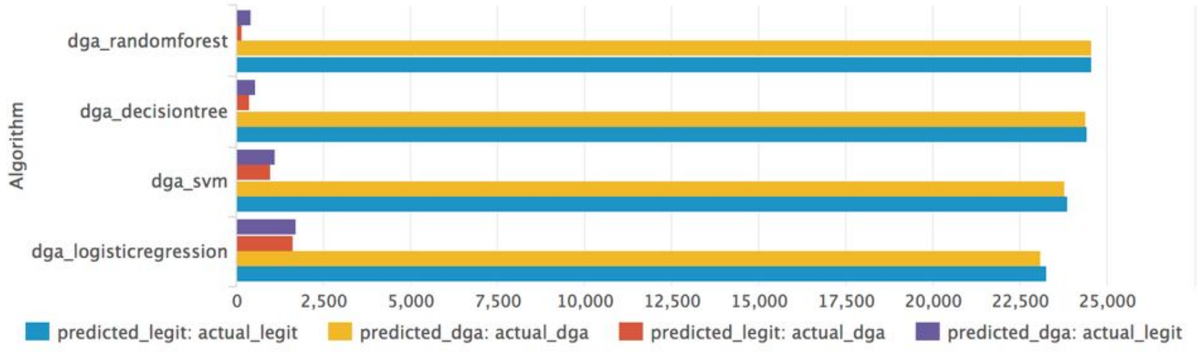
class	predicted(class)=dga	predicted(class)=legit
dga	24512	407
legit	524	24549

4. Logistic Regression

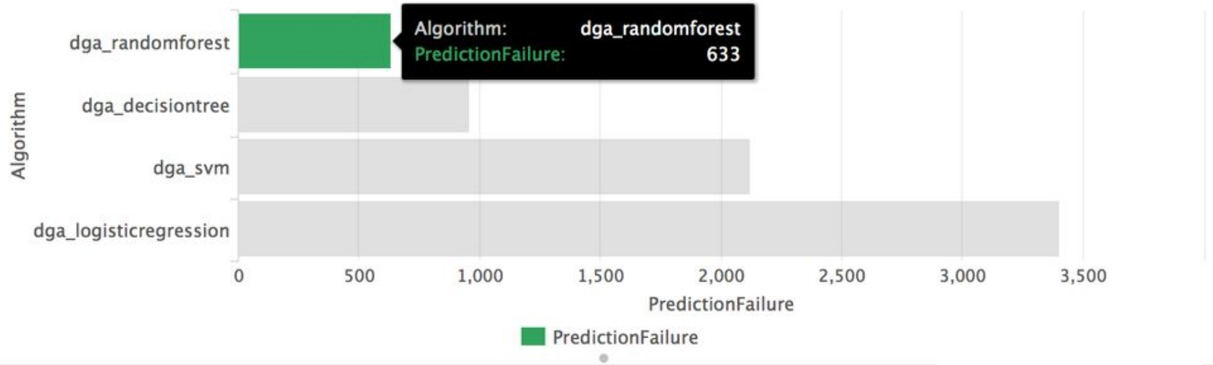
class	predicted(class)=dga	predicted(class)=legit
dga	23254	1665
legit	1686	23387

Evaluate models

Compare and evaluate models



False predictions



4. Operationalize Machine Learning

Edit | Export | ...

Machine Learning Algorithm: RandomForest

Timerange: Custom time | Submit | Hide Filters

Operationalize machine learning

As successful machine learning depends on a continuous process we constantly evaluate the results. Furthermore we can close the feedback loop and append our evaluations to our training dataset to keep models always up to date based on latest information.

Setup notes:

1. Create an index that holds domain names and computed features (we used a index named "dga_proxy")
2. Activate scheduled searches (app menu: More > Alerts) to generate sample data and fill this index.
3. Check the macro `domain_input` in Settings > Advanced Search if you have custom naming

If you want to test on your domain data please adjust your data flow to this mechanism. Of course you can also take domain name data from CIM data models in Enterprise Security and integrate into this mechanism.

Count of predictions

6

Trend of true predictions

6 -8

Trend of false predictions

0 -1

Prediction performance over time

Results of machine learning algorithm (dga_randomforest) applied to new domains

_time	domain	class	predicted(class)
2017-09-19 22:10:59	mrop8i1scak54s2hskfpq443z.org	dga	dga
2017-09-19 22:10:58	stackadapt.com	legit	legit
2017-09-19 22:10:57	cf.dropboxstatic.com	legit	legit

Results of machine learning algorithm (dga_randomforest) with DGA detected

_time	domain	class
2017-09-19 22:10:59	mrop8i1scak54s2hskfpq443z.org	dga
2017-09-19 22:10:54	8fjsofjlapjjxhcm.org	dga
2017-09-19 22:10:52	g5te08189ly791v20rrr53sdsv.org	dga

Checklist to manually adjust the results

As successful machine learning depends on a continuous process we constantly evaluate the results. Furthermore we can close the feedback loop and append our evaluations to our training dataset to keep models always up to date based on latest information.

Trend of domains classified as DGA

2,392 → 0

Trend of domains manually classified as LEGIT

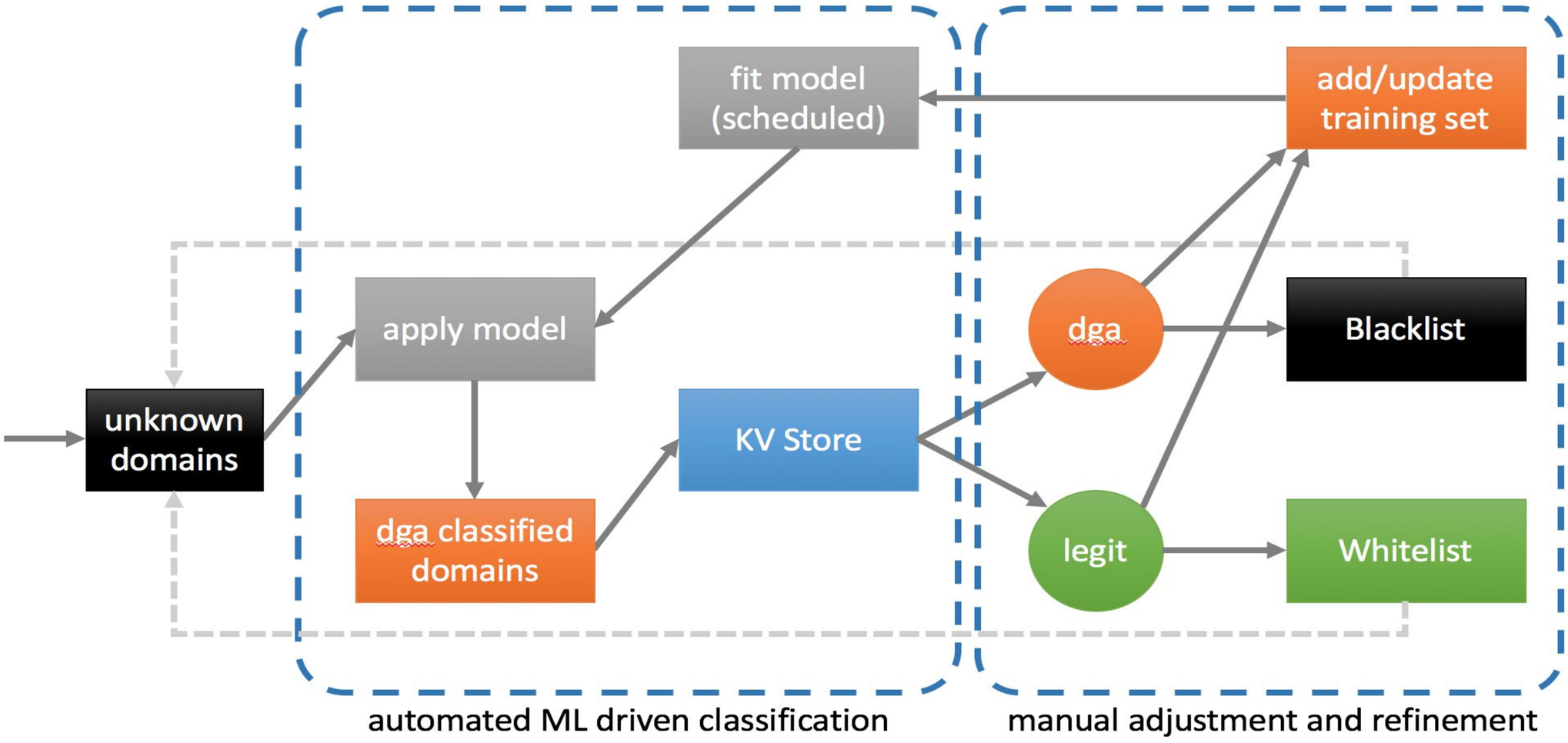
4 → 0

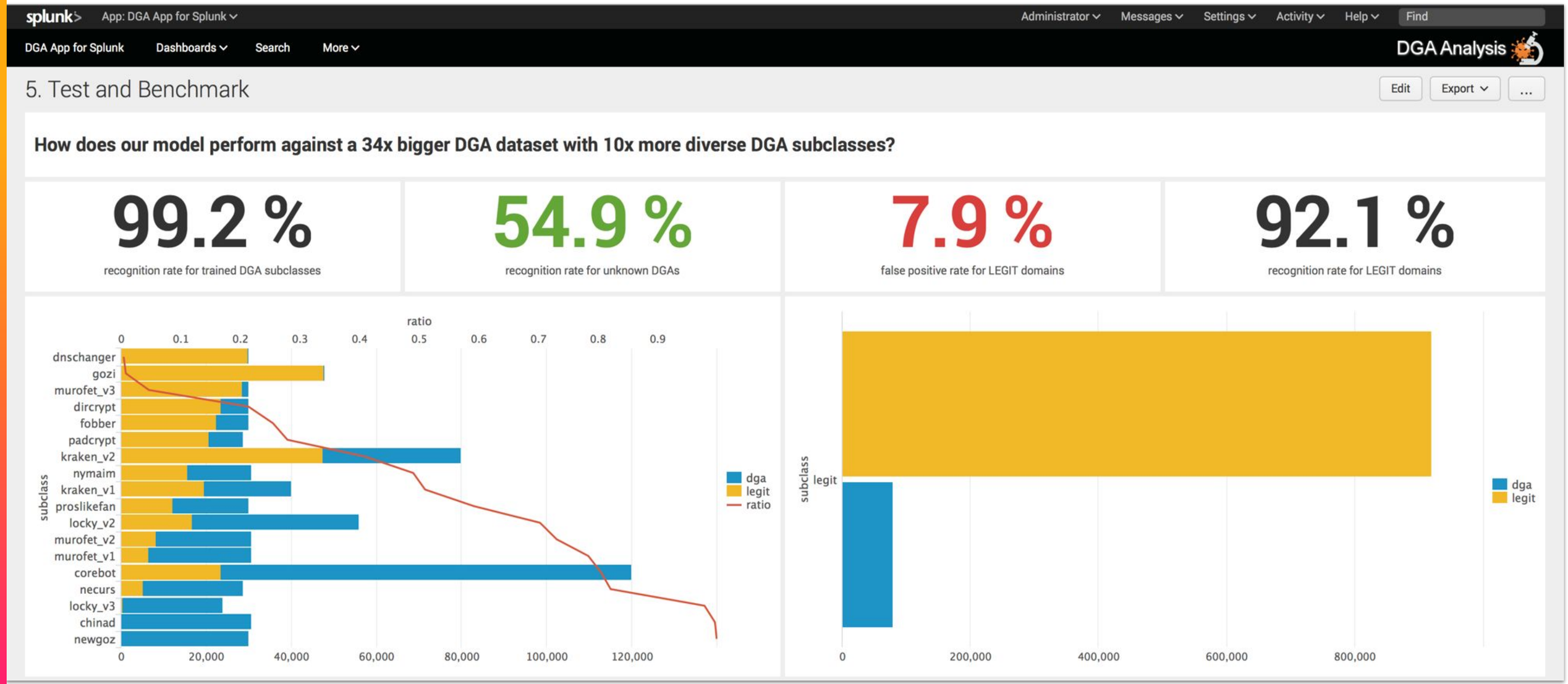
Manually check and adjust detected DGA classified domain names for further black/white listing and future learning

	time	datetime	class	domain	key_domain
1	1505852183.609000	09/19/17 22:16:23	legit	dimmhfj.xyz	LEGIT DGA
2	1505852180.980000	09/19/17 22:16:20	legit	rvxoudaurvjf.info	LEGIT DGA
3	1505852159.000000	09/19/17 22:15:59	dga	1qntxfs13eloj8hdbokd1qddfqt.org	LEGIT DGA
4	1505852157.000000	09/19/17 22:15:57	dga	nuvc6amdxse1vbtu.biz	LEGIT DGA
5	1505852154.000000	09/19/17 22:15:54	dga	qeuxctlwjmg.info	LEGIT DGA
6	1505852147.000000	09/19/17 22:15:47	dga	14fb5x4pu2zmu12eulks162u7b3.com	LEGIT DGA
7	1505852137.000000	09/19/17 22:15:37	dga	f3upm510ybndfqycfcz1ajbghu.org	LEGIT DGA
8	1505852136.000000	09/19/17 22:15:36	dga	1v31si318e57gk1gdcsi1l4t5m9.com	LEGIT DGA
9	1505852134.000000	09/19/17 22:15:34	dga	un905fm8dfb9etmx23m8sy5y.net	LEGIT DGA
10	1505852132.000000	09/19/17 22:15:32	dga	m3e3ytfvqgtj1wv1d3ka0zf3j.net	LEGIT DGA

« prev 1 2 3 4 5 6 7 8 9 10 next »

History of DGA detection and manual adjustments





- Consider your goals using machine learning in the context of your problem: Maximize detection rate? Minimize false positives?

Thank you

splunk>