Olivier Markowitch

Dimitrios Sisiaridis

19 Jun 2015

The Brufence project

Scalable Machine Learning for Automated Defence System

Automatic detection of threats and frauds in communication systems and payment transactions

Work Package 2: "Communication Systems Security - Detection of Threats and Attacks on Managed File Transfer and Collaboration Platforms"

Task 2.1: Market Intelligence

Table of Contents

Introduction

The problem

Public and private organisations deploying Managed File Transfer (MFT) and collaboration platforms based on commercial or open-source software today, have nearly no solution for predictive security analytics. Current approaches in intrusion detection tend to focus exclusively on the technical IT aspects of the detection. They detect either only existing attacks based on misuse detection (signature-based) or fail to provide a high level of protection against Advanced Persistent Threats (APTs) and zero-day attacks, based on anomalous detection. Furthermore, they do not detect efficiently attacks performed over a long period or do not cope with in an efficient way with the produced security noise and high number of false positives.
The analysis of threats and attacks demands storing of a large amount of temporal data with a high level of dimensionality with respect to spatial locations and temporal aspects of users. Big Data technologies, like Hadoop, are not easy to be implemented in practice as quite often there is a need to redesign their functionalities, especially for analytics and business intelligence services.

The proposed solution

A system for automatic detection of threats and attacks to complement communication systems to enable automatic improvements of their own defences, based on predictive analysis.

The *Brufence* system, along with exploring IT aspects will also focus on middleware, data flow and information content aspects. It will supply existing approaches by macroscopic observations of users, processes and applications. It will be based on different levels of abstraction, e.g., network interfaces, interaction with the operating system, communication level, user interaction, user behaviour and process interaction.

- It is a data-centric approach for detecting abnormal behaviour aiming to result to an autonomous self-protected communication system, by learning from past and current attacks , by detecting abnormal behaviour in its data level processes on identified *Indicators of Compromise* (IOCs), by tracing the actions done during an attack across multiple intrusion kill chains over time, and by returning a quantitative and validated estimation of the risk of future and on-going events.
- It is a proactive risk management approach using behavioural analytics based on pattern analysis of complex events regarding users, network devices and middleware. It aims to determine in near real-time the 'root-cause' of a threat, especially for APTs and zero-day attacks, by detecting behaviours that can be measured and weighted against *normal behaviour*. Reports for reducing the workload of the security analysts will be produced, through an *Alert Engine* (BAE), based on generated risk scored for events deviating by normal activity identified by *baseline profiles*, illuminating patterns and relationships for users, applications and the underlying network infrastructure. By capturing relationships between events and attackers, it will also lower the cost and time for forensics.
- It is a near real-time scalable framework for deploying Machine Learning models for predictive analytics in the field of anomalous threat detection in collaboration platforms. By adapting or rewriting Machine Learning algorithms for unstructured data, aims to enhance system functionalities for the analysis of huge amount of log data aggregating by a *Data Acquisition Engine* (BDAE), particularly in cases where there is a large amount of security noise, large dimensionality and non-stationarity of data, as well as the need for rapid and accurate identifications of threat patterns. Different models will be compared in parallel in order to maximise their efficacy in predicting *abnormal behaviour*, evaluated for their performance in terms of scalability, accuracy, readability and QoS.
-

- It is a weighted anomaly approach, combined with Machine Learning, using pattern analysis, to define indicators of threat activity. The proposed *Learning Engine* (BLE) will enable the consumption of seemingly unrelated disparate datasets, regardless of their format, to discover *correlated patterns* that result in consistent outcomes with respect to the access behaviour of users, network devices and applications involved in risky abnormal actions, and thus reducing the amount of security noise and false positives. Along with history- and user-related data, network log data will be exploited to identify abnormal behaviour concerning targeted attacks against the underlying network infrastructure as well as attack forms such as man-in-the-middle and DDoS attacks. Network connectivity and analysis of traffic data can provide valuable information to identify a single-point of attack and failure as well as the most critical or vulnerable between connected nodes, in order to improve the accuracy of the predictive model, based e.g. on graph-based classification or semi-supervised classification models.

The Brufence system is in compliance with the CERT-EU guidelines for threat detection [32][34]. It is intended to be a multi-layer, modular approach with mechanisms covering the set of actions proposed by the Guide for identification of threats and attacks, such as detecting and alerting, attack analysis, motivation identification as well as threat mitigation and refinement. Furthermore, it promotes a centralised monitoring, processing and analysis of complex events in MFTs and collaboration platforms reducing thus the impact of the visibility of data movement inside and outside (i.e., data exfiltration) the enterprise or the organisation that deploys collaboration software.

Managed File Transfer (MFT)

Key features

Managed File Transfer (MFT) software is a class of integration middleware used by enterprises as well as by public and private organizations for secure and guaranteed delivery of a file or set of files from a source to a target over a network e.g. from an organization to another directly or via a file transfer service provider or within the organization through a collaboration platform irrespectively of users' location, in order to improve operational efficiency through the automation of system-centric activities. There are three common characteristics that separate MFT software from free FTP or other types of data movement software which are *secure delivery*, *guaranteed delivery* and *auditing facilities*.

There are three different types of MFTs for the delivery of, usually, large files of multiple standards and formats: *system-centric* file transfer, *people-centric* file transfer and *extreme* file transfer.

- System-centric file transfer involves using software to automate the delivery of files between systems, or from a system to a user often by enforcing security policies using proxies and firewalls.
- People-centric file transfer or *ad-hoc* file transfer refers to the delivery of files among users through a combination of applications and collaboration platforms and tools.
- Extreme file transfer refers to the delivery of extremely large files using proprietary or enhanced protocols over UDP or in parallel over TCP with error correction facilities.

There are several factors to be considered with the movement of all data that is outbound from an organisation including encryption and ad-hoc transfer (especially, in the case of exchange large file attachments with external parties through a secure portal or a collaboration platform). Another important issue is related to the ability of stopping the movement of sensitive data, data-based matching or pattern-matching, that is known as *Data Loss Prevention* (DLP). In addition, factors related to the visibility of data movement inside and outside the organisation include centralised monitoring, notification, policy enforcements, SLA-based monitoring, centralised provisioning for all new connections, outsourcing and Third-Party managed services e.g. by using a SaaS.

Examples of commercial and open-source based MFTs

- The *Attunity Gateway* (U.S) [21] is a fully integrated solution for automated file transfer into Hadoop from any data source by providing authentication, transfer auditing capabilities and multi-tiered security architecture with DMZ front-end facilities. A DMZ (Demilitarised Zone) or, a perimeter network is used to add an additional layer of security to an organization's LAN, e.g. by using firewalls.
- The *MOVEit* MFT of the UK-based IPSWITCH company [70], can be deployed on-premise, in cloud or in a hybrid environment.
- The *Managed Information Xchange* (MIX) MFT [55], from GlobalEscape, a UK-based company, is a SaaS for secure managed file transfer in the cloud. It offers misuse intrusion detection and response services from another UK-based company, AlertLogic. The underlying delivery protocol is called *Enhanced File Transfer* (EFT) also supported by AlertLogic. They use Oracle or SQL as the underlying database for auditing while user authentication can be integrated with LDAP, Native Active Directory, ODBC, or locally.
- The Oracle's MFT [89] supports *PGP* (Pretty-Good-Privacy) encryption. PGP is a data encryption/decryption program that provides cryptographic privacy and authentication for data communication using a serial combination of hashing data compression, symmetric-key cryptog-

raphy and PKI for signing, encrypting/decrypting texts, email files, directories as well as for whole disk partitions.

- *IBM's MFTs* suite is a layered approach. It includes products such as WebSphere, Sterling and Aspera [65]. The basic components for a defence-in-depth approach include multi-factor authentication, multi-zone DMZ, session break and protocol inspection. It supports auditing controls for reporting, monitoring and delivery receipts. It offers smart automated processes such as virus scanning, integrity checking, deleting of processed files and checkpoints. It can be integrated with common security infrastructure such as Active Directory, LDAP as well key and certificate store.
- Other major vendors providing MFT solutions are *AxWay MFT* [22], *Hightail* [59], *Saison Information Systems' HULFT* [103] and *Primeur's Spazio* [95].

The above MFTs are among the most-used state-of-the-art market and open-source solutions. A competitive analysis of current MFTs can be found in [66]. All of them provide integrated authentication facilities and most of them encryption services.The MIX MFT and HULFT are the only ones providing intrusion detection, based on signatures of known threats while. None of them, to the best of our knowledge, implements anomalous intrusion detection techniques based on pattern analysis for predictive analytics.

Collaboration platforms and tools

Key features

Collaboration platforms and tools are a combination of collaborative software supporting communication, conferencing and coordination activities in enterprises as well as in organizations in public and private sector. There are many available tools supporting electronic communication, electronic conferencing, collaborative project management as well as for intra-communications and sharing of documents and knowledge.

According to *NIST* (National Institute of Standards and Technology) cloud-based computing service models are implemented either as end-user software (*Software as a Service* - SaaS), as the underlying hardware infrastructure (*Infrastructure as a Service* - IaaS), or as middleware applications in the form of a platform (*Platform as a Service* - PaaS). The latter describes the entire middleware stack between the hardware (IaaS) and the end-user (SaaS). It is very often used to enable a cloud-based business application which itself in turn is offered as a service and often referred to as a SaaS. There are two types of PaaS, the *Application PaaS* (APaaS), which provides the core technology for developing and deploying cloud applications, such as *elasticity*, or *multi-tenancy* and the *User eXperience PaaS* (UXPaaS), which provides the UI and interaction capabilities to integrate applications in the cloud with a collection of services, e.g. portal, collaboration, social, and mobile services.

Commercial and open-source solutions

There is a plethora of collaboration platforms and tools. The most well-known commercial solution is the MS Sharepoint. Alfresco [6] is an alternative, a free open-source based collaboration software. Huddle [62] is another alternative. Other solutions are either focused on collaborative management, web-content management, enterprise project management, or workflow management and information sharing.

- *Microsoft SharePoint* [81] is used by enterprises and organisations mainly to create websites. Users can store, organise and share information using their own devices through a web browser. Document libraries are stored in *OneDrive*. Collaborative team working, content management, enterprise social networking, workflow management, project management, business intelligence, web content management and information searching are among the facilities provided for authorized users. SharePoint can be installed and deployed either on premises or as a cloud-based service. *SharePoint servers* that share common resources can be grouped logically into *farms*. *SharePoint sites* can be grouped into site collections which are associated with content databases accessed through web applications. SharePoint functionality is provided via *service applications* available in a farm, integrated with *Active Directory* (AD) enabling the 'least-privileges' access requirement through a centralised management interface. External applications are allowed to give access to content databases and SharePoint capabilities to authenticated users through a proxy. Authorized users can upload 'sandboxed' plugins in a secure way.
- *Alfresco* is an enterprise content management developed in Java, for documents, web, records, images, and collaborative content development. It is an alternative to MS Sharepoint for MS Windows and Unix-like operating systems. It can be installed and deployed on premises as a free open-source software. For more scalability and modularity there are commercial open-source versions for on-premises and cloud as a SaaS. It provides authentication and authorisation based on the *Common Internet File System* (CIFS), an application-layer network protocol. Full-text indexing and searching capabilities are utilised with Apache *Lucene* search library while workflow management is implemented using the *Activity* open-source workflow engine.

- The *eXo Enterprise Social Platform* (U.S/France) [53], is a UXPaaS. It can be installed in an IaaS e.g. in *Amazon* or *OpenStack* to build a public or private cloud. Security features in eXo deal with role-based policies for users, compatible with authentication/authorization mechanisms such as CAS, JOSSO, OpenSSO and JAAS (the standard authentication and authorisation mechanism for JEE). The platform supports mobile access based on *Single-Sign-On* (SSO). It integrates with BonitaSoft's *Bonita OpenSolution* for Business Process Management to support collaboration and knowledge management features.
- *Huddle* (U.K) is another open-source based, alternative to MS Sharepoint. It has a governmental level of security. It materialises a datacenter security with storage options either in *Rackspace* servers in Europe, or in *Carpathian* servers in U.S. Data are encrypted with SSL and 256-bit AES. It implements firewalls and protected perimeter defence (DMZ), pen-testing by third-parties, DDoS mitigation services and real-time replication with 15' failover (company's estimation). File permissions are based on ACLs. The product is in compliance with UK *IL2/IL3* accreditation services for confidentiality.
- The *True Hybrid Cloud* [1], from Abiquo, a UK/Spain based company, brings together all business cloud resources into one consistent, managed platform on-premise and public clouds or for the development and deployment of a hybrid cloud. Installation is on-premises, in Abiquo's environment with the ability to to manage hypervirors (VMware ESX, Hyper-V, KVM, Oracle VM and Xen) while the options for storage are either in Abiquo anyCloud SaaS, NetApp or in public clouds such as on Amazon, Rackspace, DigitalOcean, HP and Google. For more flexibility, the company promotes ready solutions from Abiquo's partners, including *SDN* from Cohesive FT, workload mobility and scaling from CloudSoft and image building from ShareSoft, with the latter providing full data and access logging facilities.
- The *Zimbra Collaboration* [142] (U.K), is a secure open-source collaboration tool for business-class email and calendaring. It can be deployed in a private cloud, through Zimbra's service providers or in Europe and globally. It is compatible with desktop environments such as Windows, Mac and Linux. It presents flexibility for integration with other applications, such as web services in Zimbra's *AJAX* web client . The main security components in the Zimbra Security Program include the *Security Centre*, the *Security Response Policy* that utilizes the vulnerability Life Cycle workflow, the *Vulnerability Rating Classification* which utilizes NIST CVSS for scoring and communicating the characteristics and urgency of vulnerabilities and the *Responsible Disclosure Policy* for reporting vulnerabilities to Zimbra providing also a board for security news and alerts. An extra level of security can be added through Third-Party integration with trusted security open frameworks e.g. *MailGuard*.
- The *LIFERAY* [75] (U.S / China) is an open-source portal for an enterprise/organization implemented on premises, cloud (as SaaS), hybrid. It is written in Java. Its security features include identity management as well as an implementation of *OWASP*-recommended security practice. *IMB Connections* version 3.0 [67] provides social analytics to help users cut through the noise of social updates. *Yammer* [140] can be used to create a company's private social network. *ProjectLibre* [96] (U.K) is an open-source replacement for MS Project Management. *Clarizen* [38] (Israel / U.S) is a SaaS project management software. *Wrike* [139] (EU) is an all-in-one project management collaboration software, implemented on premises or cloud-based (as a SaaS) providing mobile and web-based access. Its main security features include SSO via *Google Apps* or *SAML*, support for Active Directory as well as control of users email addresses and file repositories for privacy. *SpiraTeam* [116] (U.S) is a SaaS integrated Application Lifecycle Management system for project management. *GoToMeeting* [37] (from Citrix) is a SaaS for organizing and attending of online meetings. *BaseCamp* [26] (U.S) is a web-based solution to communicate and collaborate on projects. The *WebEx Meeting Centre* [36] (Cisco) handles web online meetings, based on Cisco Spark. *Bitrix24* [28] (RU) is a SaaS for collaboration, communication, social networking, workflow and knowledge management. *Redbooth* [99] (EU) is a collaboration platform that helps teams transform company-wide collaboration. *Tamashare* [124] is a virtual secure room for collaborative activities. It lacks of any cloud storage. *SAP StreamWork* [104] is a social application focused on knowledge-worker decision

support. *SocialText* is a wiki platform, with a micro-blogging tool called 'Signal' similar to Tweeter. *Whig* keeps track on collaboration of multiple groups. *Zoho Apps* offers a variety of collaboration and sharing tools. Other collaboration tools are *Google Apps* [57], *Adobe Acrobat Connec*t and the open-source *MindTouch*.

Security considerations

The majority of collaboration platforms and tools, either commercial of free, based on open-source software, implement authentication and authorisation procedures, usually as SSO, to constrain the access to the content management and collaborative facilities. Some of them, such as Microsoft SharePoint, Alfresco, eXo, Huddle, Zimbra and True Hybrid Cloud implement also central log and identity management. Huddle is also concerned with the mitigation of DDoS attacks. There are on-premises and cloud-based implementations, some of them provided as SaaS or PaaS. Zimbra provides an advanced level of security with a focus to vulnerability sharing intelligence. There are others like eXo, Huddle, Alfresco, True Hybrid Cloud and Microsoft Sharepoint that allow third-party application integration for an advanced level of security. For example, *Titus* [128] has developed a suite for an advanced security level, for Microsoft SharePoint, focused on data classification, data loss prevention as well as data encryption. Rule-based access control policies are implemented for a secure sharing of sensitive content based on documents' metadata. Visual markings are added to identify the level of this sensitivity based on standardised security labels, promoting user awareness and accountability when handling sensitive information. For mobile users, especially for iOs devices, there is one point of access to SharePoint and cloud storage, by leveraging Microsoft Rights Management Services (RMS).

Nevertheless, although there is nowadays a trend to move directly into SaaS, and away from devices and perimeters, services such as the control, visibility and detection of threats and attacks, none of them, to the best of our knowledge, implements anomalous threat detection using predictive analytics with machine learning. According to the DARPA director [43] the problem with all information processing, including security analysis and detection for threats, is that 'the systems that are based on, are sending and retrieving information at the data output level, which leaves a 'window' for opportunities for hackers'.

BigData for event processing and predictive analytics

Key Features

BigData technologies provide mechanisms for services such the *Extraction*, *Transformation* and *Loading* (ETL) of data. *Hadoop* [58] is a distributed reliable processing framework, through *MapReduce*, and storage, through HDFS, for very large datasets, structured or un-structured. Other popular big data technologies are *Massively parallel processing* (MPP) and *NoSQL* databases implemented e.g. in document stores, Google Big Table clones, graph databases, key-value stores and data grids.

Examples of NoSQL databases are MongoDB, CouchDB, Cassandra, Redis, BigTable, HyperTable, Voldermort, Riak and Zookeeper. Other tools implementing MapReduce are Cascading, Cascalog, mrjob, Caffeine, S4, MapR, Acunu, Kafka, Azkaba and Greenplum. Examples of BigData storage operating systems are S3 and HDFS. There are a number of application servers such as EC2, Google AppEngine, Elastic Beanstalk and Heroku. The processing of large datasets can be made by using either R, Yahoo!pipes, Mechanical Turk, Solr/Lucene, Elastic Search, Datameer, BigSheets or Tinkerpop. Natural Language Processing (NLP) tools include among others the Natural Language Toolkit (NLT), OpenNLP, Boilerpipe and OpenCalais. There are several tools for profiding visualization such as Gephi, Processing Protovis, Fusion Tables and Tableau. Examples of acquisition engines are Google Refine, NeedleBase and ScraperWiki. There are several serialization frameworks such as JSON, BSON, Thrift, Protocol Buffers and Apache Avro, with schemas defined in JSON.

There are a number of tools integrated in the Hadoop ecosystem:

- *HBase* [11] is the defacto choice in working with Hadoop. It is a primary enabler to build analytic applications.
- *HIVE* [13] is a data warehouse system. It provides and SQL interface to Hadoop for batch querying, with high latency. *HIVEQL* is similar to a subject of SQL.
- *HCatalog* [12] is an API to a subset of the HIVE metastore providing a table a storage management services. It allows developers to write DDL, to create a virtual table and to access the virtual table in HIVEQL.
- There are mainly two options to achieve lower-latency queries for Hadoop, either by integrating HDFS with HadoopDB or an analytic database, where MapReduce can be invoked via SQL or, by replacing MapReduce e.g. with *Impala SQL Engine* that will run on each Hadoop node.
- *Revolution R Enterprise* [44] is a scalable, high performance platform for the rich capabilities of *R* language. It is a cross-platform integration with a wide choice of user interfaces and deployment options. The *R-Hadoop* open-source project supports interfaces to MapReduce, HDFS and HBase, while it can be run locally or remotely.
- *Spark* [18] is a framework for distributed computing, for running programmes that run in parallel across many nodes in a cluster, especially for large scale data analytics.It aims to abstract the tasks of resource scheduling, job submission, execution, tracking and communication between nodes, providing also an API to work with distributed data. It achieves high performance through caching datasets in memory, combined with low latency.
- *PIG* [16], an alternative approach to Spark, is a procedural scripting language for analysing large datasets. It consists of a higher-level language for expressing data analysis and infrastructure for evaluating these programs, including a compiler that produces sequences of MapReduce programs, for which large-scale parallel implementations already exist, e.g. in Hadoop, as well as a language layer, with a textual language called “Pig Latin”.

- *SQOOP* [19] enables bulk data movement between Hadoop and structured systems, such as RDB and NoSQL systems. By using Hadoop as an *ETL*, it acts as the E (extract) and L (load) while MapReduce handles the T (transform) of data.
- *Flume* [9] is a framework responsible to collect, aggregate and move large amount of data from different sources (e.g. web servers, application servers, mobile devices , etc.) to a centralised datastore, providing mechanisms for fault tolerance, reliability, failover and recovery.
- *Oozie* [15] is the Hadoop workflow scheduler. A client submits the workflow to a scheduler server which enables different types of jobs to run in the same workflow. It can be used also to string jobs together from other Hadoop tools including MapReduce, Pig, Hive and Scoop, using Java programs and shell scripts.
- *Solr* [17] is a highly scalable search server based on *Lucene*. It provides services such as full-text search, hit highlighting, faceted search, dynamic clustering, database integration and rich document handling.
- *Yarn* [20] is a generic resource management and application framework for deploying multiple services.
- *Graphviz* is a graph layout and visualisation package, and alternative to R visualization capabilities.

Network security and traffic log analysis

Key Features

Automated traffic log analysis is needed for advanced threat protection [108] in terms of protection on the underlying network infrastructure in secure managed file transfer and information exchange using services of a collaboration platform.

There is a need for automated traffic log analysis over a long period of time at every level of the enterprise or organisation information system including any deployed collaboration platform. Log data should be then correlated with attack communication profiles, derived from a learning set of behaviours, representing a complete picture of how an adversary acts in a variety of environments. For this purpose, machine learning algorithms can be used, for example, to examine statistical features, domain and IP reputation, Domain Generation Algorithms (DGA), which are designed to evade detection in the growing noise of web traffic in order to prevent e.g. a botnet traffic correlation. Data acquisition and data mining methods , with respect to different targeted and indiscriminate attacks, should be used (e.g., *crowdsourcing*), to get a perspective of the threat landscape. The traffic log analysis will leverage the integration of credible and actionable threat data to other security devices, in order to protect, guarantee and remediate actual threats, to get insight on how the breach occurred, thus to aid forensic investigations and to prevent future attacks.

There are inherent difficulties regarding *classified access* on content stored in a collaboration platform. Another important issue has to do with *privacy* concerns. Implemented security measures in MFTs and collaboration platforms and tools should be in compliance with EU privacy standards, guidelines and directives. Data privacy is related to topics such as privacy-preserving data publication, data mining and information retrieval. Techniques widely used are cryptography, perturbation and auditing. There are mainly categories of disclosure limitation techniques in terms of anonymising datasets for applying analytics, such as query restriction and data perturbation.

- query restriction techniques deal with controlling the size of query results, restricting the overlap between the answers of successive queries, suppressing the cells of small size and auditing queries to check privacy compromised
- data perturbation techniques refer to sampling data, swapping data entries between different cells, adding noises to the data and adding noises to the query results.

Privacy is either associated to the *physical*, *information* and *organisational level* or is related to *communications*, *behaviour* for building profiles or in *person's privacy* as e.g. for facial recognition and location tracking.The right to privacy in EU can be found in the *European Convention on Human Rights* (ECHR, 1953) [51]. The *EU Data Protection Directive* (1995) [50] incorporates the eight *OECD's principles* namely, *collection limitation, data quality, purpose specification, use limitations, security safeguards, openness, individual participation* and *accountability*. EU has developed a more comprehensive approach to privacy than U.S. Until recently, according to the '*Safe Harbor Effect*', U.S companies can transfer, store or use personal information about EU member country residents, if they meet the 'adequacy standard' of the Data Protection directive.

Privacy players can be categorised into groups such as the data collectors, data markets as the aggregators, data users and data monitors or protectors. Trusted Computing Groups such as Microsoft, Intel, IBM, HP and AMD implement *Trusted Platform Modules* based on the enforcement of *Digital Rights Management* for privacy. Moreover, *Privacy Business Models* (PBM), such as *MyPrivacy*, *MyReputation*, *SafetyWeb* and *Singly*, allow customers to find out what informa-

tion is available about them, to repair false information and to determine what information they wish to share with advertisers.

The underlying network infrastructure
Use case 1: Cisco network devices and traffic log analysis

There are certain actions that must exist in a centralised log management system in a collaboration platform, such as:

- setting up an authentication scheme to ensure synchronisation between network devices with authentication time server
- timestamps to ensure proper time format and zone for the syslog messages
- centralised logging server(s), where all syslog messages should be stored
- a login facility to log successful or failed user logins
- archive logs to log configuration changes and accounting with TACACS+ to log execution commands.

TACACS+ (Terminal Access Controller Access-Control System) is an open standard protocol developed by Cisco for handling authentication, authorisation and accounting services (AAA) which uses TCP. An alternative is the RADIUS tool (by IETF), which uses UDP. TACACS+ logs are preserved in a file managed by TACACS+ configuration settings. This file along with the syslog messages can be forwarded as inputs to the BDAE.

A centralised logging facility aims to monitor specific execution and configuration commands on a Cisco device, and feeding the Brufence system with log data for pattern analysis, in order to raise alerts to network administrators and security analysts. Some network devices have built-in scripting capabilities. They can be used by the BLE and BAE to raise an alert using these inputs i.e. internal email messages. There are several commands that should raise an alert upon execution (e.g. *gdb*, *test*, *tclsh*, *copy* and *reload*), or upon a configuration change (e.g. *service internals*, *boot* and *config-register*). Other commands that should be monitored are those that may be used to connect to line cards or switch processors, on a Cisco IOS device (e.g. *attach*, *remote*, *ipc-con* and *if-con*), commands for IOS XE (e.g. *attach* and *hw-module*), commands that can be used to the Linux shell of the Cisco IOS XE (e.g. *platform shell* and *request*), as well as those that may be used during a *reconnaissance* phase of an upcoming attack (e.g. *show*, *platform*, *show region* and *show memory*) and the '*do-exec*' version of any of the previous commands, in configuration mode. Based on the outputs of behaviour analysis and organization's SLA for access control policies on the collaboration platform or secure file exchange using an MFT, network administrators may follow an approach to restrict users access to a set of available execution commands by configuring TACACS+ to deny execution of these commands, if TACACS+ is used for AAA, specific to the used TACACS+ application.

CERT_EW proposes [33] a number or risk assessment best practices and detection methods of compromises on Cisco operating systems regarding offline modification of an IOS image by an attacker e.g. a *low level rootkit*, or runtime execution of arbitrary codee, a *high-level rootkit*. Available tools for creating rootkits are the *GDB*, the embedded GNU debugger, the *Tool Command Language* (TCL) and the IOS XE daemon '*iosd*'. GDB is present inside every Cisco network device (i.e. switches and routers). It is used for online debugging of device's operation and cannot be disabled. TCL is used for scripting functionality in IOS devices. Tclsh is enabled for accounts with privileged level 15 while several backdoors have been developed with Tclsh. The IOS XE daemon 'iosd' can be used by an adversary to gain privileged access to install a rootkit.

Use case 2: Software Defined Networks - SDNs

Software-Defined Networking (SDN) [114] is an approach to designing, building and managing networks e.g. the underlying network infrastructure of a collaboration platform. It separates the network's control and forwarding planes in order to optimise them. It offers a wide selection of competing architectures. In its simplest form, the SDN method centralizes control of the network by separating control logic to off-device computer resources.

All SDN models have an SDN *controller* that enables the network administrators to dictate to the underlying systems (e.g. switches, routers) how the forwarding plane should handle network traffic, *south-bound APIs* that relay information to switches and routers 'below', whereas the *OpenFlow* was the first standard API, and *north-bound APIs* which communicate with the applications and business logic 'above', thus, helping network administrators to programmatically shape traffic and deploy services.

- The *OpenDaylight* project [88] is a collaboration open-source project, hosted by the Linux Foundations. It aims to accelerate the adaption of SDN across the industry, for customers, partners and developers as well as to create a solid foundation for *Network Functions Virtualizations* (NFV). It supports open standards such as the OpenFlow standard.
- *OpenFlow* is a communication protocol that gives access to the forwarding plane of a network switch or router over the network. It enables remote controllers to determine the path of network packets through the network of switches. It uses ACLs and routing protocols and aims to tackle with man-in-the-middle attacks, potential single-point of attack and failure, programming and communication channel issues, as well as protocol's deployment experience. Among its benefits are that it reduces *capex* i.e. capital expenditure for the acquisition of organisation's assets, and *opex* i.e operational expenditure , delivering of agility and flexibility as well as enabling innovation.

There are several use cases implemented and deployed dealing with carrier and service providers, offering bandwidth on demand and WAN optimization, cloud and datacenters, network visualisation for multi-tenants and faster turn-around times, as well as for enterprise campuses, offering network access control and network monitoring. There are also use cases for automation and programmability such as adjusting the flows, focused on protocols e.g. OpenFlow and protocols that enable SDN controllers to interact with routers and switches in the forwarding plane while adjustments can be made as to how the traffic, flows through SDN networks, thus helping networks to respond to changing demands, as well as use cases specialized to support the applications, such as the coordination, automation and exception handling of a network to better align with the applications running on it using automatic configurations of routers and switches in a scalable manner to support rapid deployment of new applications, services and infrastructures, in compliance with organization's requirements, whereas maybe, a language can be used to generate a cross-domain response e.g. Javascript Object Notation (JSON), or the Extensible Messaging and Presence Protocol (XMPP). Finally, there are use cases deployed for automating SDN networks offering lack of any interference from a network administrator while the network decides how to address changes automatically. Examples are the *Puppet, Chef*, *SaltStack*, *Ansible*, *CFEngine* and orchestration platforms such as *OpenStack*, *VMware*, *CloudDirector,* the open-source *CloudStack and ETHANE*.

There are several threats to SDNs such as:

- forged of faked traffic flows [i]
- attacks on vulnerabilities in switches [ii]
- attacks on control plane communications [iii]
- attacks on/and vulnerabilities in controllers [iv]
- lack of mechanisms to ensure trust between the controller and management applications [v]

- attacks on/and vulnerabilities in administrative stations [vi]
- lack of trusted resources for forensics and remediation [vii]

Countermeasures to the above threats include:

- *replication* methods for threats [i, iv, v, vii]
- *diversity* for threats [iii, iv, vi]
- *self-healing* methods for threats [ii, iv, vi]
- *dynamic switch allocation* for threats [iii, iv], *trust between controllers and applications* for threats [iv, v]
- *trust between controllers and devices* for threats [i, ii, iii]
- *security domains* for threats [iv, v]
- *secure components* for threats [iv, vii]
- fast and reliable *update* and *patching* can be used for threats [ii, iv, vi].

An approach to build secure and dependable SDN by design is presented in [45]. Threat vectors are described to exploit SDN vulnerabilities and to sketch the design of a secure SDN control platform. They make use of security domains, secure components as well as fast and reliable software update and patching. Security domains are based on *sandboxing* and *virtualisation* either by isolating a rendering engine from the browser kernel or by allowing only minimal set of operations and communications between different domains. Secure components are the essential building blocks of a secure and dependable system to provide *Trusted Computing Bases* (TCB) i.e. tamper- proof devices to storing sensitive data, to assure security properties. *FRESCO* [110] is a framework for the development of security architectures in SDN.

Threat detection

Key Features

A perimeter-only security model for information sharing in managed file transfer and collaboration platforms is insufficient. With the *Bring Your Own Device* (BYOD), data now moves beyond the perimeter. According to [94] threats to the intellectual property and generally to sensitive data of an organisation, are related to *insider attacks*, *outsider targeted attacks* (i.e., Advanced Persistent Threats) and *DDoS*, or combined forms of internal and external attacks e.g. zero-day attacks or attacks performed over a long period. Adversaries can be either criminal organizations, careless employees, compromised employees, leaving employees or state-sponsored cyber espionage.

There are two types of *Intrusion Detection Systems* (IDS):

- *misuse detection* systems, based on signature-modelling of known attacks having high detection accuracy for known attacks but incapacity of detecting previously unobserved attacks
- *anomaly detection* systems, based on signature-modelling of normal traffic, with capability in detecting new attacks but usually with a high false alarm rate.

There are several categories of IDS such as *network-based*, *protocol-based*, *application protocol-based*, *host-based* (based on the analysis of the intervals of a computing system rather that of its external interfaces), *hybrid-based* and *agent-based*, with the latter, characterised by a high level of abstractions, scalability and adaptability. Common IDS products are *Snort*, *Bro*, *Suricata*, *Peakflow*, *Tripwire*, *AV* and *Mcafee HIPS*.

Misuse detection tools are simply unable to keep-up with constantly changing new attack models e.g. kill chains. One of the keys to detecting anomalous access behaviour in a collaboration platform and in managed file transfer, is to identify users, network devices and applications involved in risky abnormal actions defined in a kill chain.

- A *kill chain model*, from the side of an adversary, includes seven phases: *reconnaissance*, *weaponization*, *delivery*, *exploitation*, *installation*, *command and control* and, the *actions on objectives*.
- A series of actions as a systemic process for network defence actions are referred as *F2T2EA*: *find*, *fix*, *track*, *target*, *engage* and *assess*.
- The correspondent courses of actions in a network IDS are: *detect*, *deny*, *disrupt*, *degrade*, *deceive* and *destroy*.

Advanced Persistent Threats (APTs) are targeted attacks against specific network infrastructures, e.g., Stuxnet. They are usually based on *polymorphic malware*.

- The *Pass-the-Hash* hacking technique (PtH) is usually used in Advanced Persistent Threats (APTs) to authenticate in a remote authentication server.
- The equivalent technique that misuses Kerberos is called *Pass-the-Ticket* (PtT).

A risk management strategy that attempts to provide a solution to the detection of Advanced Persistent Threats (APTs), is presented in [63], based on the analysis of multiple intrusion kill chains over time. It makes use of three types of indicators: *atomic* (IP addresses, vulnerabilities, identifiers), *computed* (hash values, regular expressions, behavioural) and *collections* of computed and atomic indicators (e.g., statements). The method is evaluated for performance in the case of a targeted malicious email delivered to a limited set of individuals containing a weaponised attachment that installs a back-door which initiates outbound communication to a

server, utilising *SPF*. The *Sender Policy Framework* (SPF), is an email validation system framework. It is an open standard for preventing email spam and detecting e-mail spoofing, by verifying sender IP address. *SenderID* is a Microsoft email validation framework (used in MS Exchange and in MS Sharepoint) that derives from SPF, with identical syntax, based on *IETF RFC 2822*. It uses the algorithm *PRA* (Purported Responsible Address).

DDoS attacks can be categorised into:

- *volume-based*
  - they aim to saturate the bandwidth of the targeted resources.
  - their magnitude is measured in *bits/sec*.
  - examples: UDP-flood, ICMP-flood and other spoofed-packet flood.
- *protocol-based*
  - they aim to consume the actual server resources, or resources on the intermediate equipment, such as, firewalls load balancers.
  - their magnitude is measured in *packets/sec*.
  - *examples: SYN floods, fragmented packet attacks and Smurf DDoS*
- *application* layer based
  - they aim to deplete certain resources in the application by using seemingly legitimate packets.
  - their magnitude is measured in *requests/sec.*
  - send data according to specific features of well-known applications, such as HTTP, DNS, SMTP and SSL.

Examples of DDoS attacks on the OSI network model take place on the network, session and application layer:

- network attacks are performed at the Network, DataLink and Transport level such as SYN floods, connection floods, UDP floods, PUSH ACK floods, teardrop, ICMP floods, ping floods and smurf attacks
- session attacks are performed at the Transportation, Session and Presentation level, such as DNS UDP floods, DNS query floods, DNS NXDOMAIN floods, SSL floods and SSL renegotiation
- application attacks are performed at the Presentation and the Application level such as Slow Post, HashDoS, GET floods and Headless browsers attacks.

Honeypots, sandboxes and darknets

A *honeypot* is a computing resource, such as a service, an application, a system or a piece of information, isolated by any legitimate traffic, that mimics a production resource's behaviour, and whose task is to be probed, attacked, compromised, used or accessed in any other unauthorised way by a 'suspicious' entity e.g an adversary.

All the activity between a honeypot and any entity interacting with it is monitored and analysed in order to detect and confirm attempts of unauthorised usage. A honeypot should mimic a production resource in its behaviour as accurately as possible; from an attacker's point of view there should be no noticeable difference between a honeypot resource and a 'real' one.

Honeypots can be deployed for either:

- monitoring internet background noise (for scanning of worms or bots)
- learning about compromised nodes
- identifying new exploits and vulnerabilities

- capturing new malware
- studying hacker behaviour
- looking for internal infections or attacks from insiders.

Honeypots can be classified either:

- according to the type of attacked resources, as:
  - *server-side* (for network services)
    - for web-applications
      - DShield Web honeypot, Google hack honeypot and Glastopf
      - SSH honeypots such as Kippo and Kojoney
      - *SCADA honeypots* such as theCADA HoneyNet project and the SCADA HoneyNet Digital Bond
      - VoIP honeypots such as Artemisa
      - Bluetooth honeypots such as Bluepot
      - USB honeypots such as the Ghost USB honeypot
      - sinkhole honeypots such as the HoneySink
    - for general purposes:
  - *client-side* (for client applications)
- according to the level of interaction, as:
  - *high-interaction*, when the honeypot represents a real resource
  - *low-interaction*, when it emulates a resource
  - *hybrid-interaction (*a combination).

There are several open-source based honeypots:

- *high-interaction server-side* honeypots such as Argos, HiHAT, HoneyBow, Qebek and Sebek
- *low-interaction client-side* honeypots such as HoneyC, PHoneyC, MonkeySpider and Thug.
- *high-interaction client-side* honeypots such as Capture-HPC NG, Shelia and Trigona
- *low-interaction server-side* honeypots such as Amun, Dionaea, KFSensor, Honeyd, Honeytrap, Nepenthes and Tiny Honeypot.

Sandboxes are tools used for the behavioural analysis of potential malware e.g. binary executable files, documents and webpages, in an isolated physical or virtual environment to monitor possible changes by its execution, in the file system, registry, processes, loaded libraries, as well as in network traffic, in order to detect and analyse threats targeting client applications, in a similar way to client-side honeypots.

A comparison of honeypots and sandboxes can be found in [49]. The main distinctions between a sandbox and a client-side honeypot are that:

- a sandbox is more focused on a thorough analysis of the infection process and actions performed by malware
- the goal of a client-side honeypot is to identify a file as malware and optionally to identify mechanisms leading to the infection
- honeypots rarely monitor any activity taken place after the infection
- sandboxes do not allow any interaction with ‘real’ software components and run for longer periods of time than honeypots.

Darknets or *network telescopes* are networks that only observe traffic of large-scale events, including large automated scanning, worm or scanning bot activity, backscatter and impact of DDoS attacks and misconfigured network devices, without being engaged in any form of interaction.

Commercial and open-source products in threat detection

There are several commercial solutions in the field of threat detection integrated with response management, such as SIEMs (Security Information and Event Management tools) such as:

- the *ArcSight* from HP [60], *QRadar* from IBM [64], *Logrythm* [77], *RSA* and *Splunk* [117].
  - the latter is a commercial log analysis system evolved into a machine generated processing platform, often described as '*Google for usual analytics*'. It provides a huge number of analytics and data virtualisation tools and to check monitoring configuration changes while many other applications can work directly with it.
  - in Logrythm and RSA, behavioural pattern analysis is based on deploying machine learning algorithms.
  - BigData Analytics is used combined with sandboxing in ArcSight and QRadar.
- *NewRelic* and the *AppDynamics* are two other data-center commercial monitoring tools.
- *Cloud Defender* [5] is a managed cloud-based security and compliance suite SaaS offering SIEM-style capabilities, from AlertLogic, a UK-based company.
  - It provides monitoring and detection services by collecting and correlating information from customer IT environment on-premise, cloud or hybrid data-centres.
  - It analyses log data from web-applications events and network incidents to identify malicious or anomalous behaviour based on misuse detection.
  - it creates threat intelligence and incident response services based on inputs from several sources such a Honeypot Network, flow-based forensic analysis, malware forensic sandboxing, intelligence harvesting Grid, log data, asset model data as well as customer business data.
- *Cool Startup*, by Exabeam [52], a U.S-based company, is a SIEM for big data analytics with ML capabilities which enables queries to pulp internal and external threats.
- *TITUS* [128] has an inside threat detection tool designed to look at user behaviour and discover malicious behaviour integrated with their data classification/data protection solution.
- *Proofpoint*, a U.K based company offers services as a SaaS [97], for advanced threat protection with dynamic malware analysis and automated threat response using predictive analytics; among others, there is a tool specified for MS Sharepoint and Office 365.
- *Digital Guardian* [46] (a U.S based company, formerly known as Verdasys) offers services for insider and outsider threat protection, data classification and malware detection.
- *AlienVault* [7] is another U.S based company offering SIEM-style services for network security, advanced threat detection and vulnerability management along with a threat exchange platform, called *OSSIM*.
- *VASCO* [132] is an authentication company, building tools for protecting access and data to applications.

Among the most well-known open source solutions are the following:

- *Graylog* is written in Java. It uses ElasticSearch, MongoDB, and Apache Kafka. A discussion over its advantages can be found in [69].
- *Logstash*, another open-source tool, is also based on ElasticSearch. It provides many of the auxiliary features as Graylog, along with a plug-in architecture as well as a Puppet and Docker deployment support.
- *LucidWork's SiLK*, is a commercial product, based on Logstash and Apache Flume.
- Other alternative approaches:
  - using Flume, Logstash and Impala or Spark
  - using ELK, *Jira*, *Netwitness* or *Moloch*
  - using Snort or host-based detection
  - using real-time sandbox detection, as in [113]

Other products aiming to provide state-of-the-art threat and attack detection services implementing machine learning algorithms are:

- The *Darktrace Enterprise Immune System* from DarkTrace [42] , a UK-based company, is a solution for detecting unknown threats, with a clear focus on insider threats, using signal processing and machine learning algorithms, especially for critical infrastructures
  - network data, log data and user behaviour data are captured and interpreted by a data-capture real-time engine providing inputs for modelling network, humans and devices through a threat classifier that uses unsupervised learning based on recursive Bayesian estimation to model 'normal' behaviour on an organisation's corporate network.
  - any unusual or suspected behaviour can then be identified as 'abnormal' in real or near real-time
  - enhanced log data are analysed for new attack patterns and the outcome is forwarded to update the learning engine
  - raw packets of data inputs are stored for forensic analysis while the outcome of correlation analysis for detecting threats is visualised to notify security analysts for response.
- *Perforce Helix* [94] is a proactive anomalous threat detection platform for protecting intellectual property using machine learning. It is implemented using behavioural analytics in threat and attack detection
  - it compares users behaviour with their historic activities or with peers over time in order to detect and reveal indicators of attacks based on abnormal activity i.e. different from normal peer/role activity.
  - in case of outside or combined inside/outside attacks, machines are compromised with stealth malware.
  - Compromised machines or identities do not know what is 'historically' normal, therefore, they deviate and create anomalies that can be identified and raise alerts.
    - events and entities are analysed to setup behavioural event and entity scores.
    - behavioural risk scores are applied to all observed behaviours e.g. network, users and middleware, across different behavioural vectors, by using historical baselines.
    - a behavioural risk score is an aggregate of risk scores involved in the behaviour related to aggregated data as the identity, activity, asset and asset movement in a specific event.
    - entity risk models generate risk scores for users and assets by establishing baselines for all entities, such that all behaviours can be measured against normal behaviour.
    - entities are clustered together based on common behaviour. The most anomalous behaviours that involve the most important and risky entities, end up with the highest risk score.
- In the anomalous detection framework developed by Unomaly [131], data is received without parsing or normalisation, in unstructured formats.
  - a learning engine consumes data inputs regardless of format, structure and without any form of parsing.
  - it supports data formats from Syslog, SNMP and anything in plaintext over TCP/IP.
  - it produces baseline profiles based on the normal behaviour of parameters, frequencies and changes in system behaviour.
  - events are scored with a focus on the root-cause, as it is the first anomaly in the chain.

State-of-the-art EU projects on risk assessment and threat detection

- *NEMESYS* (Enhanced Network Security for Seamless Service Provisioning in the Smart Mobile Ecosystem - http://www.nemesys-project.eu/nemesys) [87] is an anomaly detection ap-

proach for mobile services. It aims to exploit and detect mitigation e.g. in case of signalling attacks. They use honeypots to attract and detect anomalies.

- *MUSES* (Multiplatform Usable Endpoint Security - http://www.musesproject.eu/Muses) [85] is a user-centric, device-dependent and self-adaptive corporate security system for the analysis in real-time of risk and trust of user actions performed with a device.
- The *RASEN* project (Risk Assessment of large scale Networked systems) [98] is an approach for security risk assessment, security testing and legal compliance. There are use cases dealing with threat scenarios using CaPEC attack patterns into the proposed CORAS risk model. A framework and a toolbox have been developed for security testing guide by risk assessment and compositional analysis.
- In the *MICIE* project (http://www.micie.eu) [80], a tool has been developed for systemic risk analysis and secure mediation of data exchanged across linked critical infrastructures. The project aims to develop an alerting system, able to identify in real-time the level of possible threats induced on a particular critical infrastructure and other interdependent critical facilities to notify the authorities providing them with a real risk level.
- The *SWEPT* project (websites through malware detection and attack prevention technologies - http://www.swept.eu - http://hdiv.org) [123] is based on the 'security-by-design' concept. It uses a set of prevention technologies adapted by the *HDIV* open-source project.
- The *TRESPASS* project (Technology-supported Risk Estimation by Predictive Assessment of Socio-technical Security) [129] takes into account IT infrastructure threats as well as threats to the human behaviour e.g. based on social engineering. An attack navigator tool has been developed to address potential attack vectors and attack paths. The tool provides an integrated risk assessment and decision support process by following different approaches to attack visualisation.
- In the field of privacy, *Kahuna* [72] is an abuseDesk solution. It is a security framework dealing with laws, industry standards, corporate policies and best practices based on Event Flow.
- The *Security Identity for protecting Business*, by Sedicii [107], is a patent technology to eliminate the storage and transmission of private user data when validating identity.
- The *Saturnus* security framework [105] is hardware-based solution for trust in mobile cloud uses *Physically Unclonable Functions* (PUF).

Threat intelligence

Key Features and best practice

According to *ENISA* [49], there is a need to go from local detection to global prevention of cyber attacks by exchanging and sharing information on incidents, i.e., threat intelligence. Enhancing functionality of existing tools demands interoperability, correlation engines for incident analysis, improved threat intelligence, advanced analytics and visualisation for massive numbers of incidents and automatic prioritisation.

There are several collaboration efforts in the field of threat intelligence:

- The *Advanced Cyber Security Centre* (ACSC) [3] is a cross-sector collaboration aiming to the protection of regions and organisations from threats. ACSC promotes the adaption of 'best' defensive strategies including FrontLine analysis, new 'predictive analytics' development as well as research and continuous development of cyber-security strategies. They propose a data-intensive cyber security monitoring framework of network packet and context-related data taking into account features such as their volume, velocity, variety and complexity, following a continuous data-flow cycle 'volume,velocity,variety,complexity'. Features related to the complexity of data include monitoring of compliance, application, user activity as well as web log data, identity and access data, and external threat intelligence. On the other hand, features related to the velocity of the data include monitoring of privileged activity, performance, transaction as well as information about vulnerabilities, configuration, change management and sensor data.
- *Threat Stream* [127] is a threat intelligence vendor. There is a free version of its threat intelligence platform called *OpticCrowd*. It is a collaboration platform that needs prior registration.
- *BringPointSecurity* (ex *Vorstack)* [30] is another threat intelligence platform for the automation, curation and collaboration of threat intelligence in cyber security by analysing, correlating and securely sharing structured an unstructured machine-readable information about current and emerging cyber threats.
- The *CAIDA-USCD* Network Telescope is a passive traffic monitoring system (i.e., a *darknet*). It collects different types of data at geographically and topologically diverse locations and makes this data available to the research community.
- *MITRE* [84] is a non-profit company, sponsored by the Office of CyberSecurity and Communications at the U.S Department of Homeland (U.S. DHS). They facilitate STIX, TAXII, CyBOX, CaPEC and MAEC in order to allow services and organisations can share cyber threat information in a secure and automated manner. CERT-EU and NATO are among the organizations contributing to the STIX discussion.
  - *STIX (*Structured Threat Information eXpression) [121] is a collaborative community-driven effort to define and develop a standardised language for representing structured cyber threat information. Implemented in python, it is an abstract data markings approach that enables marking of data down to the field level without any level of custom-marking models.
    - Current default model implementations exist for *Traffic Light Protocol* (TLP), *Enterprise Data Header* (EDH) and T*erms of Use and Simple Markings*.
    - STIX produces XML files in the form of XSD i.e. XML Schema files. A future implementation will be formal independent and including guidance for developing in XML, JSON and RDF/OWL.
    - Use cases already deployed include the analysis of cyber threats, specifying of Indicator Patterns for cyber threats, managing cyber threat prevention and response activities, sharing cyber threat information, with a detailed example use case about 'phising'.
    - STIX *profiles* can be defined to specify relevant subsets of the language. A *profile* describes which fields or values are required, are suggested, are optimal and which are prohibited. They are used to define a shared context for exactly what is expected in an

information exchange with STIX, to define a clear scope for tool/service implementation capabilities. In the current version, profiles are documented and distributed via human-readable spreadsheets while ideally, could also be machine-processable.
- *TAXII*, the Trusted Automated eXchange of Indicator Information [125], is the main transport mechanism for cyber threat information represented in STIX.
- *YETI* [141] is an implementation of TAXII written in python.
- *CaPEC*, the Common Attack Pattern Enumeration and Classification [31], is a community resource for identifying and understanding attacks. It provides a publicly available catalog of attack patterns, a comprehensive schema and a classification taxonomy.
- *MAEC* - Malware Attribute Enumeration and Characterization [78], is a structured language for malware characterization based on attributes such as behaviours, artefacts and attack patterns. It allows for the faster development of countermeasures by enabling the ability to leverage responses to previously observed malware instances.
- *CyBOX* - Cyber Observable eXpression [41] is a standardized schema for specification, capture, characterisation and communication of events and stateful properties.
  - it provides a common mechanism (structure and content) for addressing cyber observables across and among a full range of use cases including operational event management and event logging, malware characterization, intrusion detection, incident response and incident management (*IR/IM*), attack pattern characterization based on threat assessment, cyber situational awareness and digital forensics.
  - it aims to improve security properties such as the consistency, efficiency, interoperability and the overall situational awareness.
  - there are a number of utilities in GitHub as well specifications for CyBOX objects, detailed Object relationships and guidelines how to create new objects using XML Schemas and the CyBOX language.
  - the framework proposes the use of the *Event Management Automatic Protocol* (EMAP), as a standard for event management within an organisational IT environment. A detailed example for the use case of IR/IM based on STIX, can be found in [120].
- *MISP* (Malware Information Sharing Platform) [83] is a platform for sharing, storing and correlating IOCs of targeted attacks (APTs), by generating Snort/Suricata IDS rules, STIX, OpenIOC, text or cvs exports.
  - the MISP system allows to automatically import data from another system (e.g., from the Brufence system) for better and faster detection of intrusions.
  - data can be imported in various ways such as free-text imports, OpenIOC, batch import, using preconfigured templates or using custom templates.
  - when new data is added to MISP, then the system immediately can show relations with other observables and indicators by creating correlations between malware, events and attributes.
  - the system does not support the input of unstructured data. There is an open-source AGPL.

Complex Event Processing for predictive analytics in threat detection using Machine Learning

Key Features

Incident correlation refers to the process of comparing different events, often from multiple data sources in order to identify patterns and relationships enabling identification of events belonging to one attack or, indicative of broader malicious activity. It allows us to better understand the nature of an event, to reduce the workload needed to handle incidents, to automate the classification and forwarding of incidents only relevant to a particular consistency and to allow analysis to identify and reduce potential false positives.

In the Brufence project, unstructured data with respect to security-related events from users, services and the underlying network infrastructure in managed file transfer and collaboration platforms, with a high level of large dimensionality and non-stationarity as well as temporal aspects, will be correlated and classified, for predictive analytics based on pattern and behavioural analysis in order to detect 'abnormal' behaviours that deviate from 'normal' behaviour. This processing and analysis of complex events will be materialised by deploying machine learning algorithms.

Complex event processing

In *complex event processing* (CEP), *low-level* data in the form of event streams are processed to create *high-level* data. High-level data have to be available on real-time or near-real-time. Event agents for example can be used to forward events for processing e.g. by an event processing engine. Usually an event-query language is used to define rules and patterns in order to subscribe for certain event patterns. A notification engine could then be used to trigger actions performed by a security administrator.

There are several examples based on complex event processing:

- in [135], they make use of two algorithms, one based on the construction of a Bayesian network and a Monte Carlo sampling heuristic algorithm.
- a meta-model for event algebras, in Compose and Snoop, to handle streams and uncertainties in active databases can be found in [102]. There is an extension to the TelegraphCQ data stream processor to execute complex event continuous series as SQL queries.
- a homogeneous reaction rule language for complex event processing based on XML is proposed in [92].
- business process modelling and execution is integrated with CEP in [25]
- in [8] the authors propose the use of domain specific reference models for event patterns to achieve faster developing of business activity monitoring applications based on Business Activity Monitoring (BAM) and Business Process Management (BPM).
- an approach to identify suspicious, unknown event patterns in the cloud, based on real-time Discriminant Analysis, is presented in [138].

Complex event processing methods have been used recently in security management for the analysis of security-related events:

- the requirements and principles for CEP in a DBMS, rule engines and stream processing engines are presented in [122]
- in the case of RFID data management, the relevant principal that should be taken into account are presented in [90].

- a method to deal with massive audio data streams, evaluated against a Gaussian mixture modelling, is given in [4].
- the use of the temporal model in CEP is discussed in [137]
- an approach for the detection of event processing patterns in event databases is presented in [71] addressing the use of information fusion analysis
  - Information Fusion Analysis, deals with signals, features and decision level analysis over various types of data based on probability, estimation and signal processing techniques.
- one way to define trust in cyber-security is in terms of evidential reasoning using Bayesian, Dempster-Shafer and PCR5 theories. The authors in [29] propose the use of Dynamic Data-Driven Application Systems (DDDAS), based on Bayes rule, for trust in cyber-security.
  - Their approach is based on the use of metric quantification to ensure trust, in terms of the overall sum of factors such as predictability, dependability, faith, competence, responsibility and reliability.
  - A Domain Trust Authority can be used to evaluate end-to-end trust over secure communication while risk (R) is defined in terms of trust (T) as T= 1/R.
- a discussion on bare-metal analytics tools, such as Numtrace and BareCloud platform for malware analysis is presented in [126].
  - The authors propose the use of sandbox environments such as Ether, Anubis and Cuckoo, used in parallel and synchronised by a scheduler for behavioural comparison of the generated profiles, to pre-filter incoming samples.
- the *Address Space Layout Randomization* (ASLR) technique can be used to protect against buffer overflow attacks. According to [27], a way to reduce security noise would be to narrow the list of possible alerts only for persistent problems.
  - a *service ticketing system*, such as *ServiceNow* or *JIRA*, can then be used to pass the alert to an expert. *RTIR* and *Abuse Helper* are amongst the most popular tools for ticket tracking and automated incident information processing.

Machine learning for predictive analytics

Researches have argued that we need to build autonomous systems that could act in response to an attack in an early stage [68]. In the field of cyber-security and secure communication systems, intelligent machines could implement algorithms designed to identify cyber threats in real time and provide an instantaneous response with respect to their reliability, privacy, trust and overall security policy framework.

Predictive Analytics, using *pattern analysis*, deals with the prediction of future events based on previously observed historical data, by applying methods such as machine learning. For example, a supervised learning method can build a predictive model from training data. This model then is used to make predictions about new observations.

In the Brufence project for security in communication systems, machine learning algorithms will be deployed and redefined in the learning engine (BLE) by examining the efficacy of different models in order to identify abnormal behaviour in managed file transfer and collaboration platforms.

Machine learning can be classified with respect to the type of the training dataset or to the underlying theory. Thus, in the first case, there are three different types of machine learning such as:

- s*upervised learning*, by using labeled examples
- *unsupervised learning*, by using unlabelled examples
- *semi-supervised*, which is a combination

With respect to the underlying theory:

- *analytical learning* relies on domain theory or knowledge management
- *reinforcement learning* deals with the learning of a control policy
- *multi-agent learning* is an extension to a *single-agent learning*.

Other known classifications are:

- *argument-based learning*
- *interactive learning*
- *transfer learning*
- *deep learning*
- *cellular Neural Networks* (CNN)

The main features of a learning method are:

- a *description language* for probability, trees, rules, logic (first-order, propositional), equations (linear, non-linear), learning elements and performance elements
- *type of learning*
  - *batch* learning, which may be it is time-consuming while evaluation can be based e.g. on *k-fold cross-validation*
  - *on-line* learning, that may suffer from ordering effects, while evaluation can be based e.g. on *k-partitions* of the dataset using a performance curve. Examples:
    - *Naive Bayes* (NB)
    - *Stagger*, a probabilistic method
    - *Winnow*, which concepts are represented as coefficients i.e. weights of a linear equation
    - *Hoeffding tree*, a *decision tree*
    - *ensembe methods*, that make use of multiple models, such as:
      - the *Streaming ensemble algorithm*, which maintains a fixed *d-capacity* using a un-weighted collection of batch learners
      - the *Accuracy Weighted Ensemble* (AWE) which maintains a fixed-capacity of a weighed collection of batch learners
      - the *Dynamic Weighted Majority*, an extension to AWE, which maintains dynamically sized, weighted collection of on-line learning algorithms.

Deep Learning (or else known as, *hierarchical machine learning*) attempts to improve *data learning representations* (with higher level features derived from low-level features, to form a hierarchical representation) and to create models from large-scale unlabelled data using *deep neural networks*, *convolutional networks* as well as *deep belief* and *auto-regressive neural networks* for un-supervised, supervised or reinforcement and on-line learning, for pattern recognition and statistical classification.

There are a number of open-source tools for deploying machine learning algorithms:

- *Scikit-learn* [106], written in python provides libraries that can be reused for interactive workbench and to be embedded into other software for reuse, such as classification, regression, clustering, dimensionality reduction, model selection and preprocessing. It is available under a BSD licence for full open and reusable, built in NumPy, SciPy and matplotLib
- *SHOGUN* [111], written in C++, can be used also in other environments. It provides all libraries included in scikit-learn as well libraries for explorative data analysis. Its competitor is *MLpack* another C++-based ML library.
- *Accord.net Framework* [2] provides a complete framework for building Machine Learning, having a big library of ML functions, from neural networks to decision-tree systems, specialised

for computer vision and on image streams, as well as for computer audition, signal processing and statistical applications. There are available sample applications, as a start, as well as discussion groups.
- *Apache Mahout* [14] is tied to Hadoop. It aims to build a scalable machine learning library for large datasets to support several business cases. Mahout's community algorithms implemented on top of the distributed system, are available for clustering and filtering. Few of them support Spark, as they use the legacy MapReduce framework.
- *Spark mLib* [18] is usable in Java, Scala and Python. Its mLib works also for Hadoop. Its competitor is the project *MLbase*. It builds on top of mLib, where it can make queries using a declarative language.
- *H_2O* [10] is binding with Hadoop through Java. Its algorithms geared for business processes e.g. fraud, or trend predictions. They are rather not suitable for image analysis. It can interact in a stand-alone fashion with HDFS stores, on top of YARN, in MapReduce and directly in an Amazon EC2 instance.
- *Cloudera Oryx/Oryx2* [39] is designed for Hadoop. It allows machine learning models for real-time streaming. There are use cases for real-time spam filters and recommendation engines
- *GoLearn* [56], the Google's Go language, is a big collection of libraries. Data are loaded and handled in the library, since they are patterned either with SciPy, or in R.
- *Weka* [136], from Waikato University in New Zealand, provides algorithms engineered specially for data mining. There is a GNU-GPLv3 licence collection with an extendable package system with official and unofficial packages available. There is also a book available. The library was not originally aimed to be used with Hadoop although that can be realised with the means of wrappers. It does not support yet Spark, only MapReduce. It can be leveraged by Clojure users via Clj-ML library.
- *CUDA-Covnet2* [40], written in C++, is a machine learning library for neural-network applications to exploit the Nvidia's CUDA GPU processing technology. The resulting neural networks can be saved also as python pickled objects. There is support for multiple GPUs and Kepler-generation GPUs.
  - A similar project is *Vulpes*, written in F# and work with .net.
- *ConvNetJS* [118] provides neural networks libraries for Javascript. It facilitates the use of the browser as a workbench while there is a NPM version for Node.js.

Machine learning for threat detection

There are several examples in the literature of complex event processing for predictive analytics in the field of security, utilising machine learning methods. Given for example, a stream of network packets or data records, the goal is to determine network intrusions, e.g., in the underlying network infrastructure or in services provided buy a collaboration platform. This problem can be modelled as a multidimensional stream of records, containing both continuous and categorical attributes. Datasets that will be utilised by the BLE for the learning process, contain a combination of symbolic and continuous attributes. These features correspond either to the:

- basic characteristics of the connections (service, protocol, bytes transferred, etc.)
- content characteristics of the connections suggested by domain knowledge (e.g. number of IoCs and failed login attempts)
- traffic characteristics (e.g., number of connections, or the number of connections with specific kinds of errors).

A list of unsupervised multidimensional outlier detection methods that can be generalised for unstructured and unformatted data is illustrated in [76]. In some cases, a subset of the data may be labeled, and may correspond to either 'normal' behaviour, or to a new attack. Such cases can be addressed using streaming and supervised learning algorithms. Examples of supervised classifi-

cation methodologies are the *C4.5 decision tree*, *Decision Table* (DT), *Nearest Neighbour* (NN), *k-nearest Neighbour* (kNN) and *Support Vector Machines* (SVM).

Hidden Markov models and Markov chain models are used to predict network intrusions. Markov chains can predict the next state of the network data based solely on the current state; they cannot predict future states of the network data other that the next state. A way to overcome it, would be the use of *temporal pattern analysis*, a data mining technique, to find the correlative items occurring simultaneously in one transaction. There are several existing algorithms for temporal pattern analysis such as the *intra-transaction association rule mining* (based on time-stamps), *sequential patterns*, *cyclic association rules*, *frequent episodes*, *segment-wise periodic patterns* and *inter-transaction association rule mining*.

An anomaly detection method based on *time-series analysis* for network security in [112] uses the SciPy software and Apache Kafka to collect and track metrics associated with network and device behaviour.

- Kafka Topics Distributed message passing forwards either directly to Redis (in memory key-value datastore) for real-time streaming data, or through HDFS to Redis, for batch processing of historical data
- other python tools are used throughout the process:
  - Kairos, to manage the time-series DB
  - Kafka-python, to read from Kafka
  - pyspark, to read from HDFS.

The proposed approach is an inter-transactional association rule mining method, based on *Layer Divided Modelling* (LDM) for temporal pattern analysis where the time-related database is divided into sub-databases (i.e., layers) according to specified interval, determined by a set of rules in which users are interested in.

- the method searches the frequent itemisers within each sub-database using an extended FP-tree method.
- a searching strategy is employed to improve the efficiency of the algorithm that generates the inter-transactional frequent items
- performance evaluation of the method is based on the *10-fold cross-validation* process, looking at standard deviation of the classification accuracy and rule matching accuracy.
- for evaluation purposes it is compared with another inter-transactional association rule mining algorithm, the *Extended Hash-priori algorithm* (EH-Apriori)
  - the latter, takes the transactions within each sliding window as a mega-transaction based on the *Apriori* algorithm to find the frequent inter-transactional item-sets using a hashing technique to improve its efficiency.
- The proposed network IDS uses a supervised classification based on a *Collateral Representative Subspace Projection Modelling* (C-RSPM) utilising data mining and detection of sequential intrusion patterns.
- It is a proactive protection with prompt decision making and prompt for suitable actions to be taken.
- the aim is to identify *sequential intrusion patterns* in the event of an intrusion and to predict the next potentially harmful intrusions based on the rule set trained/derived from the attack history.

An unsupervised approach to identify threats using graph-based anomaly detection is presented in [47], based on data-mining algorithms. It uses the *SUBDUE* compression, an evaluation technique to discover *normative patters* i.e. non-anomalous sub-graphs.

- implemented use cases deal with modification anomalies, insertion anomalies and deletion anomalies
- the model takes into account the amount of *noise* present in the graph
- similar approaches dealing with outlier detection analysis:
  - the *anomalous subtraction detection*, based on *anomalous subgraph detection*,
  - *discovery of unusual links in a graph*
  - *analysis of removed edges*
  - methods based on *entropy*
  - *scoring of nodes* related to other nodes
  - the use of *bipartite graphs* and statistical approaches
  - measurements of the *density*, *correctness, graph's eigenvalue* and the *graph clustering coefficient*.

Threats to databases are the *privilege elevation*, by an outsider threat, and *privilege abuse*, by an insider threat. According to [23], responses to database anomalies should be based on a clear 'response policy' defined by the database security administrator. There are different approaches of response, either *aggressive* e.g. drop the packets, *conservative* e.g. raising an alarm, or *revoke privileges* e.g. based on 'privilege states'. They have proposed an approach for the detection of SQL injections utilising machine learning for defending against insider threats. An engine 'learns' the normal behaviour based on a set of profiles (a *profile* is a collection of statistical models and a mapping of models to features). The training phase consists a determination of model parameters based on the data and a threshold learning phase that calculates an anomaly score, using:

- an SQL parser, i.e an SQL query, produces a sequence of tokens
  - a *token* can be either a constant, or a non-constant
  - a *feature vector* is a vector of tokens marked as constants
    - for each constant there is a datatype which decides which statistical model should be applied to. statistical models used are the *string length*, *string character distribution*, *string structural inference*, *string prefix*, *suffix matcher* and *token finder*.
- multiple models are used to characterise user input to protect against *mimicry attacks*.

The authors have delivered a survey for database activity monitoring of research prototypes, tools and third-party commercial products. Other approaches for database security are the *DEMIDS* which is a misuse detection system for database systems, a data-mining approach for database intrusion detection presented in [61], a detection of anomalous database access patterns in relational databases in [73].

Collaborative verification and validation techniques can be used to dynamically test SOA-based systems. Given an estimated *service operational profile*, we then can determine the operational profile of the subsequent components by knowing the transitional probabilities. This *traditional probability estimation* is called the statistical phase of the service and has two steps. In the first step, which refers to the calculation of individual component reliability, a random sample of inputs from each of the component sub-domain is selected. The estimated transition probability is equal to the fraction of inputs falling into the succeeding component's sub-domain, from each of the current component's sub-domains. The second step deals with the calculation of an atomic service reliability. *System reliability* refers to the reliability of an atomic service which depends on the reliability of the interacting components, e.g., a service provided by a collaboration platform.

- a composite service, is a higher-level service, that performs tasks by composing multiple atomic services
  - it represents a new single service, often is called as a value-added service
  - its reliability depends on the type of SOA-based model chosen

A proactive monitoring approach for assessing reliability of SOA-based systems (middleware-based), using AI-reasoning on dynamically collected failure data of each service and its components along with results from random testing is presented in [35]. A Markov chain model, in this approach, from services to components, shows a probability distribution i.e. the probability of the inputs of the service to its components and the probability of the outputs from a predecessor component falling into a succeeding component. This *probability distribution* can be used to calculate the input profile for succeeding components, thus, calculating their effective reliability.:

- each service is composed of one or more components realised by components interactions
- the component system reliability can be estimated using *Markov models*, *Bayesian Reasoning* or *component dependency graphs* (directed graphs)
- predictive analytics is based on *Memory-based Reasoning* (MBR) and *Bayesian Belief Networks* (BBN)
- inputs from *test profile-based analysis* and *operation profile-based analysis* regarding reliability measures, along with inputs results from MBR and BBN, are forwarded to a *Dynamic Reliability Analysis* module
- evidence from different sources is combined, such as:
  - software process employed
  - static analysis based on the formal verification of the software products
  - evidence from the software components
- the *reliability prediction process* follows a bottom-up approach where mission-critical services provide information for failure probabilities

In the implemented near real-time monitoring system, service points are dynamically monitored for faults, thus:

- the system identifies services and components to be monitored and defines monitored parameters, duration and frequency of monitoring as well as the format and storage of the monitored data.
- data from the testing and monitoring phase are stored in the database, either:
  - based on the use of metrics as additional evidence for:
    - deterministic metric data, such as the size and complexity
    - probabilistic metric data of quality aspects such as:
      - staff's experience regarding the level of the organisation
      - the amount of time spent on component review
      - the number of changes since last release
      - the number of requirements for use-cases satisfied by this component.
  - Alternatively, we can store the number of inputs rather than the input itself:
    - either, by storing the combined data for reliability estimation based on the input specific data
      - data about each component are stored along with its input, which is subject to memory and storage constraints
    - or based on input independent data such as the component identifier, metrics, number of inputs to each of the sub-domains of the components and the failure percentage of that sub-domain.

A review of other approaches for utilising machine learning and data mining in the field of threat detection, follows:

- an approach for risk assessment with real-time constraint based on attack graphs using a feed-forward backward propagating neural network is presented in [82]. An *Early Warning System* (EWS) has been developed, similar to a SIEM, based on combined AI methods. Risk

is defined following the Matzinger's Danger Theory as: Risk=(probability x Harm)$^{(distress\_signal + 1)}$.

- *CNN* make use of *maximum Lyapunov exponents*, a kind of a circuit architecture, where the basic circuit unit is called a 'cell' (or, artificial neutron) and consists of linear circuit elements, linear transistors, linear capacitors and non-linear circuit elements with voltage-controlled resources, implemented using VLSI technique.
  - CNN has been found [93] that withstand the *brute-force* attack using *information entropy* for prediction and statistical analysis for evaluation, based on histogram analysis and correlation of adjacent pixels.
- Technologies for protection of digital images are *digital watermark* an *image encryption*. The latter can be based implemented either using *image scrambling*, *SCAN pattern*, *tree-data structures* or *chaotic systems*. All the techniques for image scrambling are hiding the statistical property of the plaintext (source image) to withstand a statistical attack, by exploiting of either a *matrix transformation*, *pixel permutation*, *affine transformation*, *magic square transformation* or *knight-tour transformation*.
  - techniques based on *SCAN* i.e. a formal language based on two-dimensional spatial accessing methodologies, can represent and generate a large number of scanning paths or space filling curves.
  - techniques based on *tree-data structures* are using image and video compression coding fields such as binary trees, quad trees and wavelet zero-trees.
- *Chaotic systems* present properties that have been widely employed to design cryptographic systems. An image encryption approach based on the use of *Chaotic Cellular Neural Network* is proposed in [93].
- A data mining technique using machine learning, that makes use of a cryptographic algorithm is proposed in [109], based on *secure multi-party computation* techniques to compute the nearest neighbours of points in horizontally distributed datasets (i.e., each party has a collection of data for the same set of attributes, but for different entities). The authors propose the use of three data mining algorithms, the *Local Outlier Factor* (LOF), *Shared Nearest Neighbour clustering* (SNN) and *k-Nearest Neighbour classification* (kNN).
  - LOF provides a quantitive measure of the degree to which a point is an outlier with high quality results in the presence of regions of different density.
  - SNN gives good results in the presence of noise, works well for high-dimensional data and can handle clusters of varying size, shape and density.
  - kNN is highly useful in medical research where the best diagnosis for a patient is likely the most common diagnosis of patients with the most similar symptoms.
- In [74], an efficient machine learning approach is introduced in order to evaluate the security level of a masked implementation of AES in the case of side-channel attacks.
- Legacy distributed and embedded systems have two layers, the *physical* and the *supervisory and control* layer. *Trust* in such systems is defined in terms of reliability and security. Typical examples are power grids, control systems and water treatment systems. In such systems, specific control applications can use different models e.g. a *Specification and Description Language* (SDL model) for representing the running systems including identical monitoring and protection modules.
  - a non-intrusive approach to enhance legacy-embedded control systems with cyber protection features in [100].
    - it is an externalised survivability management scheme, a self-adaptive approach, based on control theory, and more particularly on the observe-reason-modify paradigm, by using of an event-based control feedback loop where events and channels form the basic interfaces by which components in the framework may receive/send information.
    - there are three sub-tasks (i.e. modules):
      - 'sensing', which is the *observation module*, based on a coordination model and implemented through a Jess-based inference machine

  - 'calculating' (reasoning), which is the *evaluation module* implemented through a *Finite State Machine* evaluator and based on the *Mealy machine model* for modelling system behaviours
  - 'acting' (adapting), which is the *protection module* implemented through a multi-threaded scripting language for automatic correction, using python scripts.
- communication between the modules is done through standard interfaces such as direct calls, JNI calls and interprocess communications.
- a GUI interface permits the generation of system events with a console showing debug messages (evaluation module) along with a a response engine (protection module).
- real-time decisions are made by using *voting schemes* with a clear distinction between truthful and un-truthful voters, such as the '*out-of-n scheme*' and the '*k-out-of-n scheme*'.
- a probability distribution function is defined while sensors can be either homogeneous or heterogeneous.

There are three categories of information stored in a database, such as *identity attributes*, *quasi-identiy (QI) attributes* and *sensitive attributes*. *Disclosure limitation* techniques in data dissemination for public use try to address the linking attacks. In a *linking attack*, the adversary infers, from the micro data, the sensitive value of the victim by leveraging the association between the (QI) attributes of the victim and the corresponding sensitive value (QI-SA association). Different types of QI-SA association are the exact association which refers to the link between QI-attributes values and specific sensitivity values, using methods such as *k-anonymity*, *l-diversity* and *(a,k)-anonymity* and, the proximate association which refers to the link between QI-attributes values and a set of proximate sensitive values, using methods such as *(k,e)-anonymity, t-closeness, variance control* and *(e,m)-anonymity*.

The above models assume that the adversary possesses full identification information. In the opposite, thus with less external knowledge, there are scenarios with alternative background knowledge assumption, based on anonymization principles such as *δ-presence, (c,k)-safety, privacy skyline, m-invariance* and *sequential anonymization*.

Group-based anonymization techniques guarantee that each individual is hidden within a certain group (QI group). *Suppression* and *generalization* are two of the methodologies to achieve i.
- Generalization, for a given attribute, replaces it with a less specific, more general value that is faithful to the original e.g. by a non-informative wild card symbol '*'.
- Suppression, which is a special form of generalisation, does not release the value for a given QI attribute

Such techniques weaken the QI-SA associations, by reducing the granularity of the presentation of the QI-attributes. There are inherent difficulties such as the hardness of optimal generalisation and the curse of dimensionality which could be overcome using heuristic methods. Generally, the protection of the sensitive personal information is achieved by performing certain transformations over the micro data, that leads to the loss of the data utility, for the purpose of data analysis and data mining. Thus, it is imperative to take into account the information loss in the privacy preserving process and to optimise the data utility under the required privacy protection, especially when dealing with high-dimensional micro data. Key features of utility optimization are data utility metrics, utility-based optimisation and application-specific utility. Examples of metrics are *generalization height, average size of anonymous groups, discernibility measure of attribute values, classification metric* and *information-gain-privacy-loss-ratio*.

- *Location Privacy* requirements are inherently personalized and context-specific. They are also related to QoS. According to [134], it is inadequate to directly apply data privacy research and techniques to the problem of location privacy. Location data are extremely dynamic and subjected to frequent updates. There are two general models, the trusted service providers and

un-trusted service providers. In the first case, privacy can be preserved using a policy-based approach un-trusted service providers. In the second one, service providers usually provide countermeasures against potential privacy violation e.g. by using pseudonyms. There are identified pseudonyms insufficients as applications need for verification true identities while a user's identity can be potentially inferred from his location, either by location tracking or by external observation. A possible solution would be the use of location hiding techniques to reduce the granularity of information such as *spatial / temporal cloacking, location blurring, reporting the nearest landmark, sending false dummy locations* and *location obfuscation*. Location anonymization can be achieved using either *k-anonymity, location l-diversity* or minimum *spatial resolution* principles, by implementing a concretely centralized Trusted Third-Party model, a client-based non-cooperative model or a decentralized cooperative mobility group model.

- A privacy-oriented context-model to support the formal definition and acquisition of context descriptions based on primary context types such as the identity, location, activity and time, is presented in [119]. The authors propose the creation of a requestor and a service ontology using a three-phase context acquisition mechanism for form-filling, context detection and context extraction, using JESS rules for service inference. The proposed context-aware service architecture for services discovery includes an agent platform, a service repository and a semantic match-maker. Definitions are given in RDF/XML and OWL-S. Their hypothesis is evaluated using the *large-sample of hypothesis* for a binomial proportion.

A brief overview of currently undertaken European projects promoting the use of machine learning for predictive analytics, behavioural analysis, privacy and detection of threats and attacks, follows:

- the *MASIIF* project (Management of Security Information and events in Service Infrastructures - http://www.masiif-project.eu) [79] aims to develop a SIEM framework for service level infrastructure. The rationale behind lies to the integration of single components or combinations into existing products or services. It is a modular approach that preserves system's legacy, also with existing commercial solutions, e.g. OSSIM and Prelude.
  - The MASIIF predictive security analyser can process a behaviour analysis to detect misuse patterns.
  - There are use cases for protection against cybercrime and attacks against information systems, Critical Infrastructure Protection, fight against fraud as well as about Data Protection and Privacy, according to EU directive, using anonymization techniques, resilient storage and reliable transmission.
  - The yourCySEC tool by ATOS is an implementation of MASIIF, a service level SIEM and subject to several commercial actions with ATOS customers, used by other R&D (i.e. research and developpment) projects e.g. the FI-WARE project (Future Internet Core Platform project).
  - Other implementations are the DaMou application, by Epsilon, an electronic monitoring of physical systems, and the Nextapps, a cloud-based platform for developing web applications.
- the *NECOMA* project (Nippon-European Cyberdefence-Oriented Multilayer threat Analysis - http://www.necoma-project.eu) [86] aims to provide new means to understand cyber threats by mitigating their effect on infrastructure and endpoints.
  - It is an anomaly detection approach in backbone networks.
  - It can be used in Hadoop, for data classification and research analysis using MapReduce in order to design and build an SDN-based programmable Internet Exchange in the Internet of Things.
- the *PANOPTESEC* project (Dynamic Risk Approaches for Automated Cyber Defence - http://www.panoptesec.eu) [91] aims to design a cyber defence decision support system to support organisations in detecting attacks and responding to them. Its main features are:

  - a dynamic learning process
  - the delivering of a prototype suite for a cyber security data collection and correlation, risk quantification and assessment as well as for mitigation response prioritisation and activation
  - visualization support for security operators
  - A simulation environment has been built to mimic cyber-security sensor capabilities in operational environments such as in enterprise IT, electric power distribution an the CleanWater distribution.
- the *SPACIOS* project (Secure Provision and Consumption in the Internet Of Things - http://www.spacios.eu) [115] aims to develop and combine state-of-the-art technologies for pen-testing, security testing, model checking, automated reasoning, model inference, model extraction and automatic learning tools.
- the *VIS-SENSE* project (Visual Analytic Representations of Large Datasets for Enhancing Network Security - http://www.vis-sense.eu) [133] deals with the identification and prediction of complex patterns of abnormal behaviour in network security domain.

Threats and attacks to machine learning engines

Examples of attack mechanisms to machine learning algorithms include *buffer-overflow attacks, client-side attacks* as well as attacks deploying *cross-site scripting* vulnerabilities. Other security concerns have to do with insertion and evasion and the '*mucus availability*' attack against SNORT IDS. In the first case, the classifier could interpret packets differently than victim machine or craft packets to emphasise this difference based on packet order, time to live and repeats. In case of the '*mucus' attack*, an attacker converts SNORT ID rules into network packets, sends packets to SNORT network IDS and overloads the system false alarms.

Attacks on statistical Machine Learning include *in-discriminative attacks*, e.g., a dictionary attack, *targeted attacks*, e.g., a focused attack that is a causative to availability attack, and *pseudo-spam attacks* which are causative to integrity attacks.

A taxonomy of the attack models and strategies against the integrity and availability of machine learning techniques and systems along with possible defences against them, based on causative and exploratory attacks with a targeted or indiscriminate focus are presented in [24]. The authors argue that defences against causative targeted should be based on regularisation and randomisation techniques, while causative indiscriminate attacks can be mitigated using regularisation tools. On the other hand, exploratory targeted attacks can be handled by deploying information hiding and randomisation techniques while in case of exploratory indiscriminate attacks countermeasures should be based on information hiding. A *Hypersphere Outlier Detection* method is presented, based on a *Simple Outlier Detection Model*, that examines a causative attack to manipulate a naive learning algorithm.

Another taxonomy of attack models analysis to machine learning algorithms as well as a list of effective defences and methods to monitor attacks and modify defences to maintain performance over time are presented in [101]. According to them, an example of an effective defence is to detect attacks and their sources, by analysing adversary tools and building classifiers to detect their use or, by performing a global analysis of SPAM and malicious traffic and blacklist or shutdown malicious sites in order capture botnets that generate spam examples in controlled environments, to build a generator to create signatures, to learn templates used to generate spam and to generate classifiers to identify spam. Another way would be to deny attacker access, by denying access to training data using diversity e.g. by collecting data from geographically different sites, at different times and at different servers or by denying access to classifier design, training data and classifier decisions.

The use of robust classifiers is another strategy to enhance the security of machine learning algorithms. It can be done either by using a variety of diverse robust features such as:

- blacklists of known spammers (reputation)
- validate sender
- known spam “fingerprints” such as an image
- compare embedded URLs to URL block lists
- optical character recognition (OCR) to read text in images
- image processing to mitigate obfuscated images
- animated GIF analysis
- word counts used by Bayesian classifier
- regularised classifiers, resistant to a few bad training patterns such as tradeoff between accuracy and resistance to targeted training attacks.

It has been shown [101] that only five bad training patters added to 200 original patterns, leads to targeted deceptions. The authors propose *Clustering* and *Outlier detection* as unsupervised techniques for identifying what is abnormal behaviour based on the deviation from a normal behaviour. According to them many classifiers are robust to outliers such as *Linear SVM, MLP* and others not such as *KNN, Kernel SVM* and *Gaussian Linear Classifier*.

Classifiers should be robust to feature deletion and addition e.g. by using inexact string matching to counter obfuscation of words in spam emails, training to be robust against feature deletion or learning core elements related to attack action. They should also be able to adapt rapidly to reduce the number of security noise, e.g., by selecting a random fraction n of patterns to replace errors >= (n/1-n) or by looking over all patterns and select the most damaging fractions n of patterns to replace error >= 2n.

Measures should be taken in order to resist to the ‘*Red Herring*’ spurious feature addition, where spurious features mislead a disjunctive learner to require them for detections while the actual attacks without spurious features are initially missed or to the ‘*polymorphic Blending*’ attack, where the attacker automatically modifies attack patterns to look like normal packets by learning 1-gram and 2-gram packet statistics from normal traffic and by designing attack patterns to match these statistics, for example using vary decoder key or padding.

In [23], the *Reject On Negative Impact* (RONI) defence strategy, aims to explore the effect of contamination assumption. The approach uses a *SpamBayes spam filter* that produces a classifier. The classifier then, based on a set of spam and non-spam messages classifies new messages, either as ham (non-spam), spam or unsure.

In order to make the classifier difficult to reverse engineering, we can utilise decision and ground truth labels using a committee of multiple classifiers with randomize parameters, feature selections, training data selection as well as to forward input patterns to all classifiers and combine results before the final outputs. The adversary should not have to the actual classifier or to training data. We should not also provide feedback on the classifier decision. On the other hand, sometimes a good method is ‘to lie’ or provide random decisions. It is important to be sure that the ground truth and classifiers are correct, to use multiple unrelated sources to accurately determine ground truth for example, for spam, by using Honeypots, user labels, traffic from known malicious spammers or identical traffic sent to multiple hosts as well as to accurately determine the effect of inputs to determine ground truth by measuring the impacts and features closer to the target for intrusion detection, or by measuring the effect of packets on the victim host using accurate instrumentation and by using multiple user judgements, when available for SPAM.

Selection of tools and products - the next steps

For the needs of this research we will be focused either on commercial products or tools, that may be based on open-source software or to those that have a sufficient security level, logging, auditing and report facilities as well to provide information for pattern analysis of data streams of usually huge datasets, aggregated by the BDAE and analysed by the BLE to raise alerts by the BAE.

MS Sharepoint is the collaboration platform that meets the needed requirements. Alfresco and Huddle according to the findings are the basic alternatives to Sharepoint, having a significant share in the European market. On the other hand, Oracle's MFT with IBM's MFT suite are the most used commercial MFTs. The AxWay MFT has a significant share in the Benelux market. The MOVEit MFT, a commercial solution based on open-source code has as its advantage can be developed on premises and in the cloud. Both of them, the AxWay MFT and the MOVEit MFT are european solutions, with global acceptance.

The next step deals with the analysis of existing techniques and products identified in WP2.1. More specifically, state-of-the-art methodologies in behavioural analysis, pattern analysis and predictive analysis for users, network and applications, in MFTs and collaboration platforms, using machine learning algorithms that were explored in the previous step, now they will be analysed in detail for designing and developing auto-protection systems, focused on anomalous threat detection.

References

1. Abiquo: True Hybrid Cloud, http://www.abiquo.com
2. Accord.net Framework, http://accord-framework.net
3. Advanced Cyber Security Centre (ACSC), http://csf2012.seas.harvard.edu/Guenther-BrammerHarvardCSF25June.pdf
4. AGGARWAL, Charu C. On classification and segmentation of massive audio data streams. Knowledge and information systems, 2009, 20.2: 137-156.
5. Alert Logic: Cloud Defender, https://www.alertlogic.com/wp-content/uploads/datasheets/AlertLogic_Cloud_Defender-How-it-works.pdf
6. Alfresco: a free open-source collaboration software, https://www.alfresco.com
7. AlienVault, https://www.alienvault.com
8. AMMON, Rainer v; SILBERBAUER, Christian; WOLFF, Christian. Domain specific reference models for event patterns–for faster developing of business activity monitoring applications. In: VIP Symposia on Internet related research with elements of MIT. 2007.
9. Apache Flume, https://flume.apache.org
10. Apache H2O (UK), http://h2o.ai/#/
11. Apache HBase, http://hbase.apache.org
12. Apache HCatalog, https://cwiki.apache.org/confluence/display/Hive/HCatalog
13. Apache HIVE, http://hive.apache.org
14. Apache Mahout, http://mahout.apache.org
15. Apache Oozie, http://oozie.apache.org
16. Apache PIG, http://pig.apache.org
17. Apache Solr, http://lucene.apache.org/solr/
18. Apache Spark mLib, https://spark.apache.org/mllib/
19. Apache SQOOP, http://sqoop.apache.org
20. Apache Yarn, http://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-site/YARN.html
21. Attunity MFT for Hadoop, http://www.attunity.com/products/attunity-mft-rmft
22. AxWay: MFT, https://www.axway.com
23. BARRENO, Marco, et al. Can machine learning be secure?. In: Proceedings of the 2006 ACM Symposium on Information, computer and communications security. ACM, 2006. p. 16-25.
24. BARRENO, Marco, et al. Can machine learning be secure?. In: Proceedings of the 2006 ACM Symposium on Information, computer and communications security. ACM, 2006. p. 16-25.
25. BARROS, Alistair; DECKER, Gero; GROSSKOPF, Alexander. Complex events in business processes. In: Business Information Systems. Springer Berlin Heidelberg, 2007. p. 29-40.
26. BaseCamp, https://basecamp.com
27. BigPanda: White Paper, http://techcrunch.com/2014/10/27/bigpanda-wants-to-bring-order-to-it-alerts-madness/?ncid=txtlnkusaolp00000602
28. Bitrix24, https://www.bitrix24.com

29. BLASCH, Erik; AL-NASHIF, Youssif; HARIRI, Salim. Static Versus Dynamic Data Information Fusion Analysis Using DDDAS for Cyber Security Trust. Procedia Computer Science, 2014, 29: 1299-1313.
30. BringPointSecurity/Vorstack, https://www.brightpointsecurity.com
31. CAPEC: Common Attack Pattern Enumeration and Classification, http://capec.mitre.org
32. CERT-EU: CERT-EU White paper- 2014-011-v1.0-06/01/2015, http://cert.europa.eu/static/WhitePapers/CERT-EU-SWP_14_07_PassTheGolden_Ticket_v1_1.pdf
33. CERT-EU: CERT-EU-SWP - Cisco IOS / IOS XE operating systems mitigation, v1.5, http://cert.europa.eu/static/WhitePapers/CERT-EU-SWP_14_08_CISCO-Risk-Mitigation_1_5.pdf
34. CERT-EU: DDoS Overview & Incident Response Guide - July 2014, http://cert.europa.eu/static/WhitePapers/CERT-EU-SWP_14_09_DDoS_final.pdf
35. CHALLAGULLA, Venkata UB; BASTANI, Farokh B.; YEN, I.-Ling. High-confidence compositional reliability assessment of SOA-based systems using machine learning techniques. In: Machine Learning in Cyber Trust. Springer US, 2009. p. 279-322.
36. Cisco: WebEx Meeting Centre, http://www.webex.com/products/enterprise_meetings.html
37. Citrix: GoToMeeting, https://www.gotomeeting.com
38. Clarizen, http://www.clarizen.com
39. Cloudera Oryx/Oryx2, https://github.com/cloudera/oryx
40. CUDA-Covnet2, https://code.google.com/p/cuda-convnet2/
41. CyBOX: Cyber Observable eXpression, http://cybox.mitre.org
42. Darktrace, http://www.darktrace.com
43. DARPA, online article: http://www.govtech.com/dc/articles/DARPA-Director-Calls-for-Cybersecurity-Change.html
44. Delivering value from BigData with Revolution R Enterprise and Hadoop, http://www.revolutionanalytics.com/revolution-r-enterprise: Executive White Paper, 2013
45. Diego Kreutz, Fernando M.V. Ramos, and Paulo Verissimo. 2013. Towards secure and dependable software-defined networks. In Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking (HotSDN '13). ACM, New York, NY, USA, 55-60.
46. Digital Guardian, https://digitalguardian.com
47. EBERLE, William; Holder, Lawrence; Cook, Diane. Identifying threats using graph-based anomaly detection. In: Machine Learning in Cyber Trust. Springer US, 2009. p. 73-108.
48. EngGame, online presentation, Time-series analysis for network security, http://www.slideshare.net/mrphilroth/scipy2014
49. ENISA: Detect, SHARE and Protect: Solutions for improving threat data exchange among CERTs - Oct. 2013, Technical Report, 2003, https://www.enisa.europa.eu/activities/cert/support/information-sharing/detect-share-protect-solutions-for-improving-threat-data-exchange-among-certs
50. EU: EU Data Protection Directive (1995), http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:31995L0046&from=en
51. EU: European Convention on Human Rights (ECHR, 1953), http://www.echr.coe.int/Documents/Convention_ENG.pdf
52. Exabeam: Cool Startup, https://www.bizety.com/2014/07/09/cool-startup-exabeam/

53. Exo: the eXo Platform - Enterprise Social Platform (U.S / France) [45]: http://www.exoplatform.com
54. Fülöp, L. J., Tóth, G., Rácz, R., Pánczél, J., Gergely, T., Beszédes, Á., & Farkas, L. (2010, July). Survey on complex event processing and predictive analytics. In Proceedings of the Fifth Balkan Conference in Informatics (pp. 26-31).
55. Global Escape: Managed Information Xchange (MIX), http://www.globalscape.com
56. GoLearn, https://github.com/sjwhitworth/golearn
57. Google: Google Apps, https://www.google.co.uk/intx/en/work/apps/business/
58. Hadoop [29.1]: https://hadoop.apache.org
59. Hightail: MFT, https://www.hightail.com
60. HP ArcSight, http://www8.hp.com/us/en/software-solutions/siem-security-information-event-management/
61. HU, Yi; PANDA, Brajendra. A data mining approach for database intrusion detection. In: Proceedings of the 2004 ACM symposium on Applied computing. ACM, 2004. p. 711-716.
62. Huddle (U.K) [52]: https://www.huddle.com
63. HUTCHINS, Eric M.; CLOPPERT, Michael J.; AMIN, Rohan M. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. Leading Issues in Information Warfare & Security Research, 2011, 1: 80.
64. IBM QRadar, http://www-03.ibm.com/software/products/en/qradar-siem
65. IBM's MFT suite: WebSphere, Sterling, Aspera, http://www-03.ibm.com/software/products/en/category/managed-file-transfer
66. IDC, Technical Report: 'A Competitive analysis on MFT software', http://idcdocserv.com/252028.pdf
67. IMB: IBM Connections, http://www-03.ibm.com/software/products/en/conn
68. InfoSec Institute, Cyber security and Artificial Intelligence: a dangerous mix, http://resources.infosecinstitute.com/cybersecurity-artificial-intelligence-dangerous-mix/
69. Infoworld, online article: 'Open source Graylog puts Splunk on notice', http://www.infoworld.com/article/2885752/log-analysis/open-source-graylog-puts-splunk-on-notice.html
70. IPSWITCH MFT: MOVEit, http://www.ipswitchft.com/moveit-managed-file-transfer
71. J. Mihaeli and O. Etzion. Detecting event processing patterns in event databases. 2007. URL: http://www.dcs.bbk.ac.uk/ptw/vldbw07/edaps/mihaeli.pdf.
72. Kahuna (Netherlands), https://www.kahuna.nl/kahuna-abusedesk-solution
73. KAMRA, Ashish; TERZI, Evimaria; BERTINO, Elisa. Detecting anomalous access patterns in relational databases. The VLDB Journal, 2008, 17.5: 1063-1077.
74. LERMAN, Liran, et al. A machine learning approach against a masked AES. In: Smart Card Research and Advanced Applications. Springer International Publishing, 2014. p. 61-75.
75. LIFERAY, http://www.liferay.com
76. Lior, Rocach, Ben Gurion University of the Negev, When cyber-security meets ML , a presentation at Penn State University, http://fr.slideshare.net/liorrokach/cyber-securityshort
77. LogRythm, https://www.logrhythm.com
78. MAEC, http://maec.mitre.org
79. MASIIF: Management of Security Information and events in Service Infrastructures, http://www.massif-project.eu
80. MICIE, http://www.micie.eu

81. Microsoft: SharePoint, https://products.office.com/en-us/sharepoint/collaboration
82. Mihai-Gabriel, Ionita, and Patriciu Victor-Valeriu. "Achieving DDoS resiliency in a software defined network by intelligent risk assessment based on neural networks and danger theory." Computational Intelligence and Informatics (CINTI), 2014 IEEE 15th International Symposium on. IEEE, 2014.
83. MISP: Malware Information Sharing Platform, http://www.misp-project.org
84. MITRE (USA), http://www.mitre.org
85. MUSES: Multiplatform Usable Endpoint Security, https://www.musesproject.eu
86. NECOMA: Nippon-European Cyberdefence-Oriented Multilayer threat Analysis, http://www.necoma-project.eu
87. NEMESYS: Enhanced Network Security for Seamless Service Provisioning in the Smart Mobile Ecosystem, http://www.nemesys-project.eu/nemesys/
88. OpenDaylight project, http://www.opendaylight.org
89. Oracle: MFT, http://www.oracle.com/us/products/middleware/soa/managed-file-transfer/overview/
90. PALMER, Mark. Seven principles of effective RFID data management. Progress Software, 2004.
91. PANOPTESEC: Dynamic Risk Approaches for Automated Cyber Defence, http://panoptesec.eu
92. PASCHKE, Adrian; KOZLENKOV, Alexander; BOLEY, Harold. A homogeneous reaction rule language for complex event processing. 2007.
93. PENG, Jun; ZHANG, Du. Image encryption and chaotic cellular neural network. In: Machine Learning in Cyber Trust. Springer US, 2009. p. 183-213.
94. PERFORCE HELIX: White Paper: Threat detection for protecting intellectual property: security and risk analytics, http://www.perforce.com.
95. Primeur, Spazio MFT,: http://www.primeur.com
96. ProjectLibre, http://www.projectlibre.org
97. Proofpoint, https://www.proofpoint.com
98. RASEN: Risk Assessment of large scale Networked systems, http://www.rasenproject.eu
99. Redbooth, https://redbooth.com
100. REN, Shangping, et al. A non-intrusive approach to enhance legacy embedded control systems with cyber protection features. In: Machine Learning in Cyber Trust. Springer US, 2009. p. 155-181.
101. Richard, Lippman, MIT Lincoln Laboratory, a Jones seminar presentation at Dartmouth College, http://engineering.dartmouth.edu/events/using-machine-learning-to-improve-security-in-adversarial-environments/, 14/01/2011.
102. RIZVI, Shariq. Complex event processing beyond active databases: Streams and uncertainties. 2005. PhD Thesis. Master's thesis, EECS Department, University of California, Berkeley.
103. Saison Information Systems,The "HULFT" Series: Securely Realizing Any Intra- or Inter-Company Data Collaboration,, http://home.saison.co.jp/english/products/hulft.html
104. SAP StreamWork, http://www.sap.com/pc/analytics/business-intelligence/software/streamwork
105. Saturnus security framework, https://www.intrinsic-id.com/products/saturnus/

106. Scikit-learn, http://scikit-learn.org/stable/
107. Security Identity: protecting Business (UK/Italy - Sedicii), http://www.infosecurityeurope.-com/__novadocuments/86662?v=635672837438500000
108. Security Week, online article, 'Automated traffic log analysis: a must have for advanced threat protection', http://www.securityweek.com/automated-traffic-log-analysis-must-have-advanced-threat-protection
109. SHANECK, Mark; KIM, Yongdae; KUMAR, Vipin. Privacy preserving nearest neighbor search. In: Machine Learning in Cyber Trust. Springer US, 2009. p. 247-276.
110. SHIN, Seungwon, et al. FRESCO: Modular Composable Security Services for Software-Defined Networks. In: NDSS. 2013.
111. SHOGUN, http://www.shogun-toolbox.org
112. SHYU, Mei-Ling; HUANG, Zifang; LUO, Hongli. Efficient mining and detection of sequential intrusion patterns for network intrusion detection systems. In: Machine Learning in Cyber Trust. Springer US, 2009. p. 133-154.
113. Skizzle Sec: Security and Computer Things, 'Security analysts discuss SIEM's Elastic-Search, LogStash, Kibana vs. Arksight, Splunk and more', http://skizzlesec.com/2014/06/08/security-analysts-discuss-siems-elasticsearchlogstashkibana-vs-arcsight-splunk-and-more/
114. Software-Defined Networking (SDN), https://www.sdxcentral.com/resources/sdn/what-the-definition-of-software-defined-networking-sdn/
115. SPACIOS: Secure Provision and Consumption in the Internet Of Things, http://www.spa-cios.eu
116. SpiraTeam, https://www.inflectra.com/SpiraTeam/
117. Splunk [29.12]: https://www.splunk.com
118. Stanford University: ConvNetJS,Technical Report, http://cs.stanford.edu/people/karpathy/convnetjs/
119. Stephen J. H. Yang; Jia Zhang; Angus F. M. Huang. Model, properties and applications of context-aware web-services. In: Machine Learning in Cyber Trust. Springer US, 2009, p. 323-358.
120. STIX IR/IM, http://cybox.mitre.org/documents/Cyber Observable eXpression (CybOX) Use Cases - (ITSAC 2011) - Sean Barnum.pdf
121. STIX: Structured Threat Information eXpression (USA) [25.1]: http://stix.mitre.org
122. STONEBRAKER, Michael; ÇETINTEMEL, Uğur; ZDONIK, Stan. The 8 requirements of real-time stream processing. ACM SIGMOD Record, 2005, 34.4: 42-47.
123. SWEPT: Websites through malware detection and attack Prevention Technologies, http://swept.eu
124. Tamashare, http://www.tamashare.com/en
125. TAXII: Trusted Automated eXchange of Indicator Information, http://taxii.mitre.org
126. The Register online: Vxer fighters get new stealth weapon in war of the malware, http://www.theregister.co.uk/2014/08/18/vxer_fighters_get_new_stealth_weapon_in_war_of_the_malwares/
127. Threat Stream, https://www.threatstream.com
128. Titus, http://www.titus.com

129. TRESPASS: Technology-supported Risk Estimation by Predictive Assessment of Socio-technical Security, http://www.trespass-project.eu
130. TSAI, Jeffrey JP; PHILIP, S. Yu. Machine learning in cyber trust: security, privacy, and reliability. Springer Science & Business Media, 2009.
131. Unomaly, http://unomaly.com
132. VASCO, https://www.vasco.com
133. VIS-SENSE: Visual Analytic Representations of Large Datasets for Enhancing Network Security, http://www.vis-sense.eu
134. WANG, Ting; LIU, Ling. From data privacy to location privacy. In: Machine Learning in Cyber Trust. Springer US, 2009. p. 217-246.
135. Wasserkrug, S., Gal, A., Etzion, O., & Turchin, Y. (2008, July). Complex event processing over uncertain data. In Proceedings of the second international conference on Distributed event-based systems (pp. 253-264). ACM.
136. Weka, http://www.cs.waikato.ac.nz/ml/weka/
137. WHITE, Walker, et al. What is next in event processing?. In: Proceedings of the twenty-sixth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems. ACM, 2007. p. 263-272.
138. WIDDER, Alexander, et al. Identification of suspicious, unknown event patterns in an event cloud. In: Proceedings of the 2007 inaugural international conference on Distributed event-based systems. ACM, 2007. p. 164-170.
139. Wrike, https://www.wrike.com
140. Yammer [50.13]: https://www.yammer.com
141. YETI, https://github.com/TAXIIProject/yeti
142. Zimbra Collaboration, https://www.zimbra.com