Computer Science Review 40 (2021) 100371

Contents lists available at ScienceDirect

Computer Science Review

journal homepage: www.elsevier.com/locate/cosrev

Review article

# Vulnerability retrospection of security solutions for software-defined Cyber–Physical System against DDoS and IoT-DDoS attacks

Manish Snehi *, Abhinav Bhandari

*Department of Computer Science and Engineering, Punjabi University, Patiala, Punjab, India*

ARTICLE INFO

*Article history:*
Received 30 June 2020
Received in revised form 15 January 2021
Accepted 19 January 2021
Available online 5 February 2021

*Keywords:*
Cyber–Physical System
Internet of Things
IoT
Software-defined networks
SDN
Fog Computing
Distributed Denial of Service
DDoS
IoT-DDoS

ABSTRACT

The wide dispersion of the Internet of Things (IoT), Software-defined Networks and Cloud Computing have given the wings to Cyber–Physical System adoption. The newfangled society relies so much on Cyber–Physical Systems, such as Smart Cities, Smart Agriculture, Medical Cyber System, that a dearth to any of the available services may lead to severe concerns. The IoT devices are unwittingly contributing to the denial of service attacks. Though the neoteric Software-defined Anything (SDx) paradigm has offered effective solution approaches to catastrophic IoT-based DDoS attacks, the novel designed solutions confront various vulnerabilities due to less secure IoT devices, high-volume real-time network traffic generated by the colossal amount of IoT devices, etc.

In this paper, we present a comprehensive survey on vulnerability analysis of security solutions for Software-defined Cyber–Physical System. The paper delineates the architectural details of the Software-defined Cyber–Physical System and recommends amalgamation of Fog Computing as one of the architectural layers for overcoming a number of vulnerabilities. As contemporary technologies like IoT, Software-defined Networking and Cloud Computing are the soup ingredients of the Software-defined Cyber–Physical System, each of the individual components has been auscultated individually for security vulnerabilities with a focus on Distributed Denial of Service (DDoS and IoT-based DDoS) attacks. To anticipate the future recasting of the novel paradigm, we discuss the ongoing research and detailed vulnerability analysis with a focus on resiliency, performance, and scalability. Last but not least, we discuss the lessons learned and prospects to conclude.



**Contents**



* Corresponding author.
*E-mail address:* snehi.manish@outlook.com (M. Snehi).

https://doi.org/10.1016/j.cosrev.2021.100371
1574-0137/

## 1. Introduction

In recent years, the pervasiveness of technology has reached a new level and will continue growing like a chain reaction. The presence of technology is evident in almost every domain. The neoteric Cyber–Physical System is a fusion of physical and cyber world glued together with network backbone for communication and feedback loops. The Internet era has enabled the physical world (like sensors in Medical devices, Agriculture, etc.) to get connected to cyber systems with the introduction of enabling technologies like physical sensors, Internet of Things (IoT), and Cloud Computing [1,2]. Due to a reasonable amount of efforts invested in IoT, the state-of-the-art IoT technology has matured and several de facto standards are on the stack for use. With the advent of IoT devices, Industry 4.0 brought a convergence of IoT devices, enterprise network, and control system infrastructure in the industry together [3]. Internet of Things (IoT) has enabled common objects to go intelligent and communicable.

The networking system has always been an inseparable component of any system where communication is involved between system entities. The transport and control protocol inside routers have enabled the information packet inside digital packets to move around the world. Though the traditional embedded network boxes that include tightly coupled control and data plane are widely adopted, traditional networking hardware is complex, less resilient, and hard to manage. Software-defined networking (SDN) [4] has proven the backbone by offering cost-effective and smart networking solutions to the grid of technologies [5,6]. It breaks the tight coupling of control and data planes into two independent entities and moves the control plane to centralized controllers. The novel architecture simplifies configuration management and policy enforcement [7]. The SDN paradigm has offered users the ability to program the devices through centralized controllers, provides controller of the big picture of the underlying network, easy configuration, and reduced cost.

Another critical component of the Cyber–Physical System is "Cyber World". Cloud Computing forms the basis for Cyber World. Cloud computing, having its roots in the 1960s when the idea of offering computing as a service was conceived, has its emergence and wide-adoption since 2007 [8]. Cloud computing has been proven the convergence of computational capacity, mobile communication, and the use of the internet. Cloud's offerings

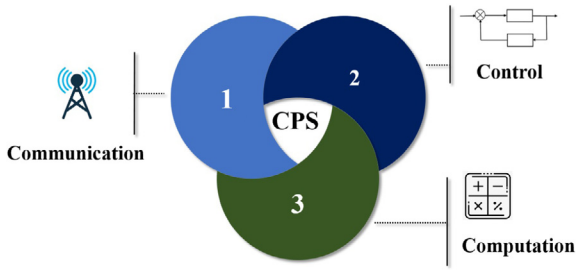


**Fig. 1.** Cyber–Physical System elements.

such as 'pay-as-you-go', elastic resources (e.g. Amazon's EC2 instances [9]), broad network access, pooling of resources have brought the cloud computing in lime-light [10].

The interconnection of the physical world, the cyber systems, and enabling technologies has given rise to the complete eco-system known as Cyber–Physical System (CPS) [11,12]. A Cyber–Physical System is comprised of computation, communication, and control elements (As shown in Fig. 1) coupled with a feedback system [12]. The fusion of Software-defined networks in the communication layer transforms Cyber–Physical System into ***Software-defined Cyber–Physical System*** (SD-CPS) and has a potential considerable influence on the digital society. The functional components of CPS consist of (a) connecting the physical and cyber world though advanced networking techniques, (b) intelligent data processing, and advanced data analytics.

As more and more of internet-enabled devices are connecting to cloud services, cloud services are flooded with high volume data. Though cloud infrastructure has high computing resources and provides prominent storage for data in abundance, it introduces latency to the edge network. Fog Computing [13] introduces a layer of distributed computational and storage resources between the perception layer and cloud layer and provides cloud-like services near to the edge devices. Fog Computing promises conservation of network bandwidth by reducing the network traffic traveling to the cloud, cost reduction, improved response time, congestion control, and enhanced security near to edge devices [14].

As the proverb says, "Everything comes with a cost", the same is true for the innovation and development in technology. On

one side, the technology has produced remarkable results to modern living and on the other side, the growth and omnipresence of technologies have turned up several securities issues and challenges to the techno-social era. The most devastating security attacks are Distributed Denial of Service attacks which compromises the system availability. DDoS attacks are coordinated and performed by a large number of distributed devices, a.k.a. bots, by sending an enormous amount of attack traffic to victim [15], thereby, making the system unavailable for legitimate requests. The geographically distributed and overwhelming IoT devices make the effects of DDoS manifold. IoT devices are most vulnerable and can easily be compromised [16]. The Mirai [17,18] and other botnets are a wake-up call to apply better security to IoT devices.

In an attempt to make machines intelligent, advanced machine learning [19,20] and deep learning [21] concepts are pressing priority. Big Data analysis [22,23] and other novel techniques are recommended in conjunction with the Fog Computing approach to deploy DDoS/IoT-DDoS security solutions. The trending algorithms are suggested to attain low-latency industrial cyber–physical systems [24] a.k.a. Industry 4.0 standards and smart systems.

The spread of the Cyber–Physical System to include diverse technologies makes it indispensable to perform an extensive study of security solutions to uncover the vulnerabilities of all the individual ingredients of the ecosystem. Hence, a deep dive into the pool to perform vulnerability analysis is a need of the hour.

In this paper, we aim to deliver a detailed taxonomy of work in the domain of Software-defined Cyber–Physical Systems with a focus on DDoS and IoT-based DDoS attacks. The prime objective of the study is to deliver comprehensive vulnerability analysis to form a basis for an ideal solution against IoT-based DDoS attacks.

### 1.1. Our contributions

The primary contributions of the review to the research community are as under:

#### 1.1.1. Most comprehensive study to date

In this paper, we have presented, to the best of our knowledge, the most comprehensive study on the security of Software-defined Cyber–Physical Systems (with a focus on DDoS/IoT-DDoS). The details include architectonics of incessant building blocks of a typical Software-defined Cyber–Physical System. We believe that the listed contributions are exhaustive and include all the important findings in the emerging domain to date.

#### 1.1.2. Profound insights into the software-defined cyber–physical system

We have presented the logical and organized view of the Software-defined Cyber–Physical System and security solutions in the pertinent discipline as below:

(i) *Detailed Vulnerability Analysis* has been carried out based on the literature survey.
(ii) *Study of sub-systems in individuality:* The study has not been limited only to "Software-defined Cyber–Physical System in the mass", but the security solutions against DDoS attacks for IoT devices (the Perception Layer), Clouds (the Cyber Layer), and Software-defined networks (the communication backbone) have also been studied in individuality as they are the jig-saw pieces of the Software-defined Cyber–Physical ecosystem.
(iii) *The Fog Computing Layer* has been proposed for effective communication and overcoming the performance issues in a typical Software-defined Cyber–Physical System.

#### 1.1.3. Most exhaustive literature survey

Literature survey is more exhaustive and elaborate as described below:

(i) *The experimentation testbed setup and implementation details* have been included in the literature survey, wherever possible. The detailed presentation scheme will prevent the reinvention of the wheel. The researchers can focus on the problem statement and leverage the infrastructure details from other researchers in a similar area.
(ii) *A number of mash-ups for SDN and/or Fog Computing solutions for DDoS/IoT DDoS mitigation and/or defense* have been studied in the literature survey to have a wide-angle view of the problem area. The solutions can, therefore, be applied across Horizontal and/or Verticals of the problem area.

#### 1.1.4. The case study from industry leader (AWS):

We have also presented a case study from Amazon Web Services (AWS) for DDoS Mitigation in the Cloud environment. As the cloud is an essential, inseparable element of a CPS system for time-critical calculations, storage, and feedback system, hence, a primary target for DDoS attacks to dump the whole system. It is the only managed layer in a CPS eco-system that has been standardized to offer security implementation against network breaches. AWS offers security features that, undoubtedly, deserves the researcher's attention.

### 1.2. The quality of research

The journals and reports from high repute databases have been considered as the primary source of information:

1. ScienceDirect, (https://www.sciencedirect.com/)
2. IEEE xplore digital library, (https://ieeexplore.ieee.org)
3. Springer Link (https://link.springer.com/)
4. NETSCOUT World Wide Infrastructure Report, (https://www.netscout.com/report/)
5. KrebsOnSecurity, (https://krebsonsecurity.com/)

### 1.3. Paper organization

The rest of the paper is organized as follows. Section 2 describes the Software-defined Cyber–Physical System and its components. The due attention has been given to the communication layer as it forms the basis for characterizing network traffic for building effective solutions. Furthermore, the section elucidates the challenges and security threats in the Software-defined Cyber–Physical System.

Section 3 describes the Fog Computing concept and its architectural details. The Fog layer is proposed in SD-CPS for improving the performance of solutions designed for mitigating high-volume DDoS/IoT-based attacks.

Section 4 focuses on the most devastating Distributed Denial of Service attacks (DDoS) and IoT-based DDoS attacks that impact system availability.

Section 5 forms the core of the survey and presents a comprehensive view of the on-going research efforts and associated challenges.

Section 6 presents the cloud case study for DDoS attacks mitigation from the market leader, Amazon Web Services.

In Section 7, the literature survey is analyzed and vulnerability analysis is carried out based on the literature survey. The section summarizes the open research issues and challenges.

The discussion in Section 8 concludes the paper and proposes future scope and challenges.

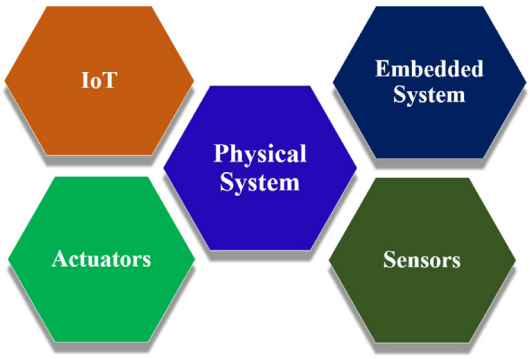


**Fig. 2.** The Physical System and IoT.

## 2. The software-defined cyber–physical system

Software-defined Cyber–Physical System (SD-CPS) is a blend of modern-day technologies such as Physical Systems and Internet of Things (IoT) a.k.a. 'The Perception Layer', Cloud Computing (The Cyber Layer), and Software-defined Networks (The Communication Backbone). The current section places the light on the SD-CPS components, its attributes, the reason for the prevalence and growth of the SD-CPS, and the architectural details of these inter-related sub-systems that offer their services in collaboration to produce remarkable results. Special attention has been given to the networking layer as: (1) it is the most vulnerable layer of an SD-CPS, (2) If the architecture of SDN is leveraged, most of the attacks can be preventing from happening.

### 2.1. Physical systems and Internet of Things

The semi-conductors have undergone an astonishing transformation over the spatial and temporal scale. It has completely altered the way we see the world. The seamless integration of transistors, sensors, actuators, and other electronics components, huddled into tiny boards, is nevertheless surprising, has led to the inception of intelligent embedded devices and it is the greatest thing since sliced bread. Fig. 2 describes the collaboration of cutting-edge components hitting the road. The evolution in the semiconductor industry has made physical systems an essential part of almost every larger system. The state-of-the-art Internet of Things (IoT) is one of such inventions that has broken new grounds. IoT is an integration of sensors, network components, and computational elements [25]. It has offered a set of new services for the technological innovation wave. An unspoken goal of IoT is to go through radical and transformational changes leading to Cyber–Physical System i.e. when physical and cyber components are intertwined for information exchange and control of physical systems through feedback loops [11].

### 2.2. Cloud computing

The world is moving towards the Internet of everything, computing infrastructure is another example. The enterprise service providers are migrating computing resources, storage, development-deployment platforms, and software to centralized data centers and offering on-demand services to end-users. The services are offered in the form of infrastructure, platform, and software. The 'pay-as-you-go' model has popularized the cloud platform and users/enterprises are adopting cloud in the form of private, public, community and/or hybrid deployment models [13,26]. The cloud has made the presence of computational resources and software availability possible all over the globe. Therefore, the cloud has become an end-destination for data from IoT devices. Furthermore, the virtualization-based on containers (such as Dockers [27], Linux Containers [28], CRIU [29], etc.) has offered a lightweight migration. Hence, Cloud has been promoted as an essential and inseparable element of SD-CPS.

### 2.3. The communication layer

The network is the backbone of any communication system. The availability of any of the end-points in a communication system is highly dependent on the underlying network. Hence, the communication layer is the most vulnerable component of SD-CPS. Most of the cybersecurity solutions are leveraging the network layer for the detection and mitigation of security attacks. Hence, the following section provides the detailed architecture of the Software-defined communication layer.

#### 2.3.1. Software-defined networking: An analogy

Software-defined networking (SDN) is the paradigm shift from traditional IP networks towards centralized software-based network control with an intent to decouple the Control Plane (Network Brain) and Data plane. The Open Networking Foundation (ONF) which is a nonprofit consortium aimed to provide the SDN development and standardization defines SDN as networking architecture that decouples the control and data planes thereby introducing a layer of abstraction between network infrastructure and applications [7]. SDN has a strong momentum to influence technology giants like Google, Yahoo!, Facebook, Microsoft, etc. The Software-defined networking (as shown in Fig. 3) aims to focus on [30–32]:

(a) Breaking the vertical integration of the control plane from the underlying data plane (routers and switches).
(b) Control logic implementation in the centrally located controller, thereby providing a larger view of the network.
(c) Exposing the Application Programming Interfaces (APIs) in the control plane as well as in the data plane.
(d) The APIs exposed to application developers are high-level interfaces known as Northbound APIs. Similarly, a set of instructions for forwarding devices is defined by Southbound APIs. The southbound interface defines the protocol between the control plane and data plane elements. Some of the protocols for Southbound interfaces are OpenFlow, ForCES, Protocol Oblivious Forwarding (POF) [33].
(e) Controller capability expansion by means of programmable interfaces. Network applications leverage the Northbound APIs offered by the control plane to extend the controller functionalities including (but not limited to) firewalls, load balancers, etc.

There are a number of open source as well as commercial versions of SDN controllers available in the market. ONOS [34,35], Ryu [36], OpenDayLight [37,38], POX [39] are some of the open-source SDN controllers, while HPE VAN SDN Controller offered by Hewlett Packard [40], Agile Controller 3.0 by Huawei [41] are subset of the available enterprise solutions.

In 2011, market leading companies including Google, Microsoft, Facebook, and Juniper laid the foundation of non-profit consortium for network infrastructure standardization and have achieved an association of 200 members [42,43]. Open Networking Foundation, With the advent of SDN, more and more organizations are moving towards software-defined networks. Tech-giants like Google [44], CISCO [45] and VMWare [46] etc. are providing winds to SDN wave. Consequently, many more enterprises ranging from network providers to network hardware manufacturers to Cloud service providers are adopting the SDN wave.

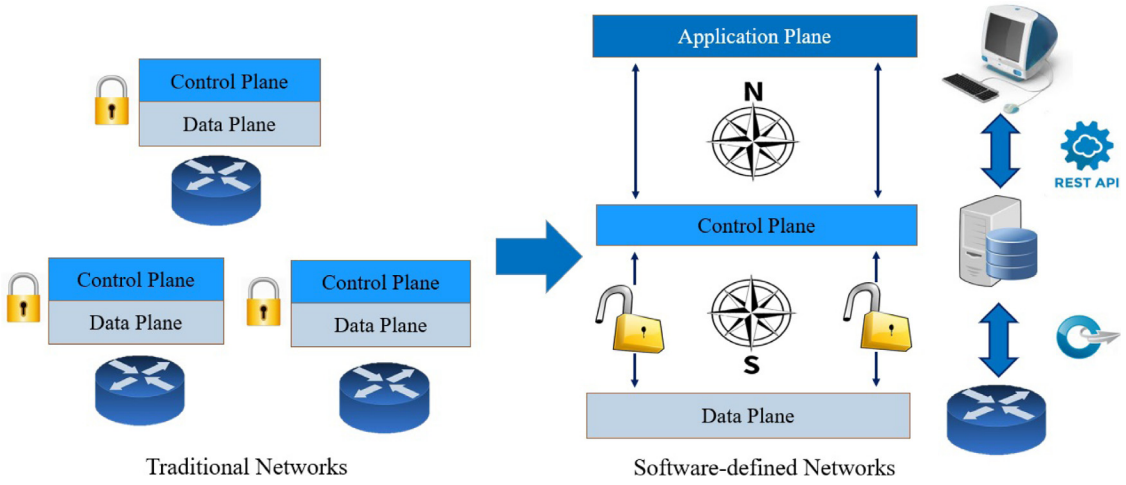


**Fig. 3.** Software-defined Networking — An Analogy.

**Table 1**
Southbound Interface protocols.

| Protocol | Typical usage |
|---|---|
| OpenFlow [47] | De-dacto protocol between SDN controller and dataplane switch |
| NETCONF [48] | Is recognized as configuration and management protocol for virtualized and extensible networks |
| OFDP [49] | Is used as network discovery protocol in SDN controller |
| OF-Conf [50] | Is a companion protocol for OpenFlow to configure and remotely manage OpenFlow switches |
| OVSDB [51] | Is a configuration protocol mainly used for open virtual switches |
| YANG [47] | Is a modular tree-based data modeling language ideal to be used along with a configuration protocol like NETCONF |

*2.3.2. Data transmission protocols in SDN*

The controller exposes well-defined APIs in order to communicate to the data forwarding plane as well as the application plane [32]. The interfaces towards forwarding/data planes are referred to as the Southbound interface (SBI). The SBI is used to program network devices and network device uses it for reporting the network status. In a standard SDN architecture, Southbound interfaces exist between SDN controller and network devices. OpenFlow and NETCONF are the widely adopted standards at Southbound interface (More standards for Southbound interface are discussed in Table 1).

Similarly, the interfaces exposed by controller towards the application plane are referred to as Northbound interfaces (NBI). The application plane uses NBI to program the SDN controller according to the needs of application which in turn is translated by the controller and programmed on to network devices. A set of RESTful APIs [52], Java APIs are the first preference of network programmers for utilizing Northbound interfaces.

*2.3.3. Open flow forwarding in SDN*

The Open Flow enabled switches are capable of operating in proactive and reactive modes. The process of SDN controller operating in reactive mode, i.e. installing the flow rules in switches dynamically as the network packets arrive, is referred to as "OpenFlow Forwarding" [53]. The sequence of steps involved in OpenFlow Forwarding is illustrated in Fig. 4. For simplicity reasons, we have considered two OpenFlow switches (S1 and S2, each with two ports: Port-1 (P1) and Port-2 (P2)), two hosts (H1 as source host and H2 as destination host) and ARP as the type of request for OpenFlow understanding. The series of steps are as follows:

STEP 1. Source host (H1) initiates the communication by sending the packet to Port-1 of switch-1 (S1, P1).
STEP 2. The OpenFlow switch, upon receiving the packet, processes the default rule for the new-flow i.e. forward the packet to the controller through Southbound interface if the flow entries are unknown. The message type for the packets received by the controller is PACKET_IN. At this point, the controller runs the L2 Learning Switch application and stores the entry i.e. Mac-address for H1 is found on (S1, P1).
STEP 3. The controller, however, does not know where the H2 is, so it sends the message back to S1 asking it to flood the packet on all ports but P1. The message type for packets sent by the controller to switches is PACKET_OUT.
STEP 4. S1, upon receiving the PACKET_OUT, sends the packet to (S2, P2).
STEP 5. S2 repeats the steps for the received packet.
STEP 6. S2, upon receiving PACKET_OUT from the controller, sends the packet to (S2, P1).
STEP 7. Since it is an ARP request, H2 replies with the unicast frame as it, at this point, knows the Mac-address of H1.
STEP 8. S2 still does not know what to do with the packet. OpenFlow flow entries are unidirectional. Hence, flow entries need to be programmed in both directions. So, the frame is sent to the controller as PACKET_IN message. The controller, however, now knows that the Mac-address of H2 is mapped to (S2, P1). The frame is sent back to S2 for forwarding. Additionally, the controller can update the flow entries of both the switches (S1 and S2).
STEP 9. Any traffic that follows the same route afterward, is forwarded directly without involving controller.

*2.3.4. Network drift towards SDN*

The adoption trend of the market is leaning towards SDN technologies. The reasons for its wide acceptability are manifold due to its cost-effectiveness, quick implementation, remote upgrades, reduced deployment and service efforts, smaller outage due to centralized and virtualized environment, extensible and scalable architecture, and capability to control the network from a centralized location.

Moreover, as the networks are expanding, most of CPS systems, cloud service providers, and Internet Service Providers (ISPs) are showing their interest in software-defined networks. The obvious reasons are benefits offered by SDN technology such as cost-effectiveness, scalability, manageability, etc. The global SDN market is forecasted to grow at a compound annual growth

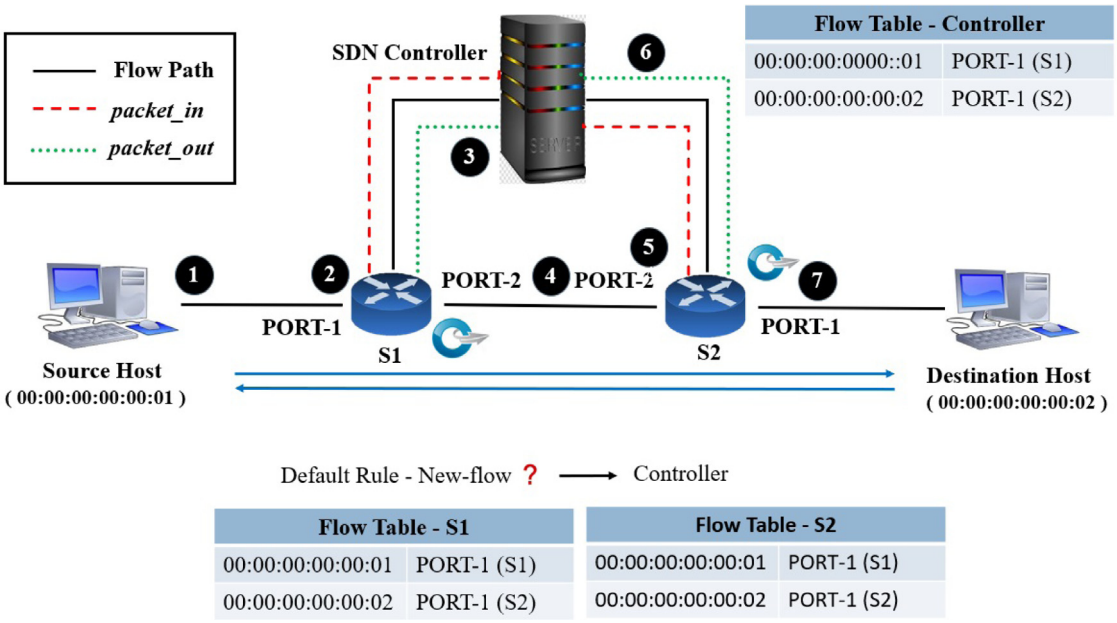


**Fig. 4.** OpenFlow Forwarding.

rate (CAGR) of 42.41% from 2018 to 2023. The global SDN market is expected to exceed a valuation of $59K M by 2023 up from $7,211.5 M in 2017 which reflects an excellent growth rate [54].

Another driving technology that has supported SDN popularity is its integration with Network Function Virtualization (NFV) [55,56]. SDN and NFV have a close relationship, though SDN and NFV are independent technologies, both advocate the application of open software and networking hardware [57,58]. For implementing network function as Network Function Virtualization, a virtual machine is created for each of the network function that needs to be migrated to NFV [59]. Though NFV can be applied to any network element, some of the areas that have shown significant adoption to NFV are switching elements, mobile network nodes, home routers, set-top boxes, tunneling elements, traffic analysis nodes, policy servers, AAA servers, cache servers, load balancers, firewalls, virus scanners, intrusion detection systems, spam protectors [60].

### *2.4. Cyber–physical systems: An overview*

The unprecedented growth in physical objects which are getting connected to the Internet has been proven a boon to introduce the idea of the Internet of Things (IoT). Though the trends like the Internet of Things can be traced in the early 80s by connecting Coca Cola vending machine at Carnegie Mellon University to the internet [61]. Similarly, cloud computing has been into existence since 2000 [62]. The innovation in network domains has proven a boon to establish a connection between the physical world (E.g. Cars, Smart Watches, Medical Devices, Sensor Elements, etc.) and cyberspace. Interdependency and integration of Internet of Things (a.k.a. IoT which represents the physical world) and cloud computing (which represents the cyberspace) have given birth to an era of Cyber–Physical Systems (CPS). The term CPS was first introduced by Gill in 2006 at National Science Foundation (NSF) to package physical and cyber entities together under an umbrella [63]. Cyber–Physical System is a genius piece of engineering architected in a layered model. A CPS system comprises of communicable, controllable, computational physical elements connected to cyberspace [12]. Most Cyber–Physical systems adhere to 5C architecture i.e. Smart Connection, Data-to-Information conversion, Cyber, Cognition, and Configuration (Feedback/Action).

In summary, Cyber–Physical Systems are a confluence of physical systems (viz. Distributed Sensors, Mobile Systems, Embedded and/or Real-time systems) coupled to cyberspace to form a CPS ecosystem [64]. Cyber–Physical Systems entails infrastructure of data acquisition, data aggregation, integration, data processing, and control system for feedback loop [65]. CPS systems are not just a group of embedded systems or real-time systems, they are far more superior to existing control systems or embedded systems because these systems integrate the physical and cyber system and make the system: (1) interactive with cyber capabilities in the included physical systems, (2) Dynamically reconfigurable, (3) scalable (4) automated with closed control loops and (5) intelligent because the systems can learn and adapt [66]. The physical objects are controlled by the cyber system based on the data analysis and inferences generated as a result of data analysis.

Though IoT and CPS systems appear to be similar in the view that both the systems are designed to increase the connectivity between the physical world and cyberspace, they have the obvious differences. IoT emphasizes interconnecting the physical world entities, thus it is an open platform; the CPS system focuses on interconnecting physical entities, information exchange, and controlling the physical systems with a loop-back mechanism (In Fig. 5) [67]. In a sense, IoT is a subset of CPS.

The CPS systems are an integrated part of modern society. One can witness CPS technology into (almost) every domain ranging from transport, energy, environment, medical, social networking, agriculture, manufacturing, telecommunication and a lot more areas yet to surface and explored. Some of the cyber–physical systems are control systems, smart cities, smart grid applications [68], connected cars, connected trains, medical cyber–physical systems [69], etc.

The implementation of the cyber–physical system in medical and healthcare is a critical system with distributed but networked, context-aware medical devices [70]. For example, *Medical Cyber–Physical System* in the cardiovascular medical domain that includes different stages of a cyber–physical system such as data acquisition, data processing, cloud computing, and actionizing [71]. The medical data is captured by a gateway from a medical device built with sensors and transmitted to the Application Server or Cloud platform in Hospital through the internet.

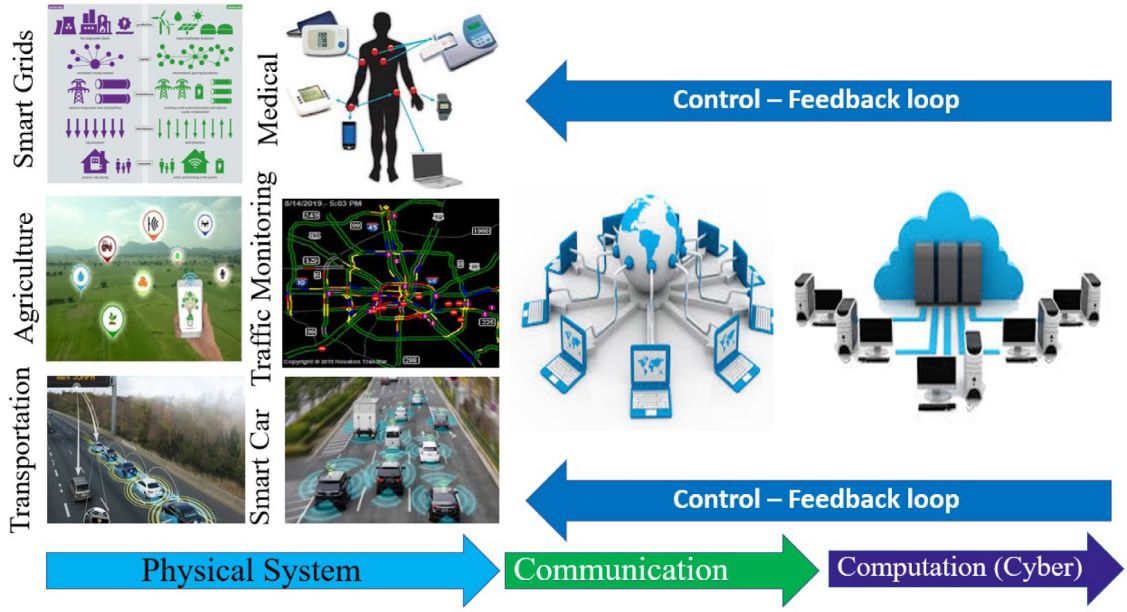


**Fig. 5.** The Cyber–Physical System Eco-System.

The data analysis algorithms monitor the incoming data and in case of any threshold violation, SMS or email alerts are generated so that timely action can be taken by clinician or doctor [72,73]. Hence, it is a loopback system with feedback.

Another implementation of the Cyber–Physical System is the most prevalent *Industry 4.0 in production engineering*. Industry 4.0 aims to deliver transparency and real-time control in the production control system [74]. The Industry 4.0 machines are augmented sensor nodes, network connectivity, and cyber system to visualize and monitor the production line and implements the variations with the feedback system. The Industry 4.0 architecture implementation comprises the smart contracts for data communication, data analysis to extract the information, accomplish computations, collaborate, perform diagnostic and decision making at cyber layer and self-optimization, self-configuration for resilience to close the feedback loop [75].

#### 2.4.1. Software-defined cyber–physical systems

The popularization of Cloud computing, one of the ingredients of the Cyber–Physical System, and the Communication network for CPS would not have been possible with the support of advanced network techniques and modern networking hardware. In the absence of SDN, CPS providers were supposed to build and deploy more and more data centers containing thousands of network switches and other hardware. Combining cyberspace with SDN paradigm i.e. Software-defined Cyber–Physical System (E.g. Fig. 6 shows one of the implementation of Software-defined Cyber–Physical System i.e. Software-defined Smart Agriculture) has opened gates of new research opportunities to closely integrate the application hosting in cloud with the programmable network [76,77].

### 2.5. Factors influencing cyber–physical systems growth and popularity

There are several factors influencing the growth and popularity of the CPS system follows:

(a) Exponential increase in IoT devices and Cloud Computing platform.
(b) A wide range of application areas such as smart agriculture, smart cities, smart homes, traffic monitoring, etc.
(c) Support for advanced network infrastructure such as Software-defined networks.

Networking is the backbone of the Cyber–Physical System as it decides how well the physical and cyber systems are connected in terms of ease of configuration, deployment, maintenance, and expansion of networking hardware. In the modern digital era, the Internet is proliferating to provide services to users by hosting software, services, and even the infrastructure to the cloud. Software and Services are hosted on to the cloud in the form of Software-as-a-service (SaaS), underlying platform as Platform-as-a-service (PaaS), and Infrastructure as Infrastructure-as-a-service (IaaS) [62]. With the advent of trending technologies viz. Mobile Computing and Internet of Things, (almost) everything is connected to the Internet and are major data providers for the cloud platform. Fig. 7(a) shows the market growth trend of IoT devices in terms of IoT connected devices worldwide [78] and 7 (b) shows public cloud adoption trend (2019) by enterprises [79]. By 2025, public clouds are expected to contain 49% of the world's stored data [80].

The network growth is exponential; nature is highly heterogeneous and expansion is extremely complex. Even a small change in the network policy leads to a sequence of ripple effects that are hard to implement, time-consuming, and very expensive. The network operators are required to be highly proficient in order to manage the traditional networks because of tight vertical integration of the control plane and data plane into a single data forwarding device (Router, Switch, etc.). Therefore, traditional networks are hard to reconfigure, not adaptable to a fault and load changes, vertically integrated and resistant to expansions. Software-defined networking has proven a catalyst in the growth of Cyber–Physical Systems.

### 2.6. Challenges in software-defined cyber–physical systems

Although software-defined CPS shows a number of benefits, it faces numerous challenges that must be taken care of while designing the CPS and includes handling enormous traffic data, performance challenges, connectivity and availability, scalability and security issues [66,77,81]:

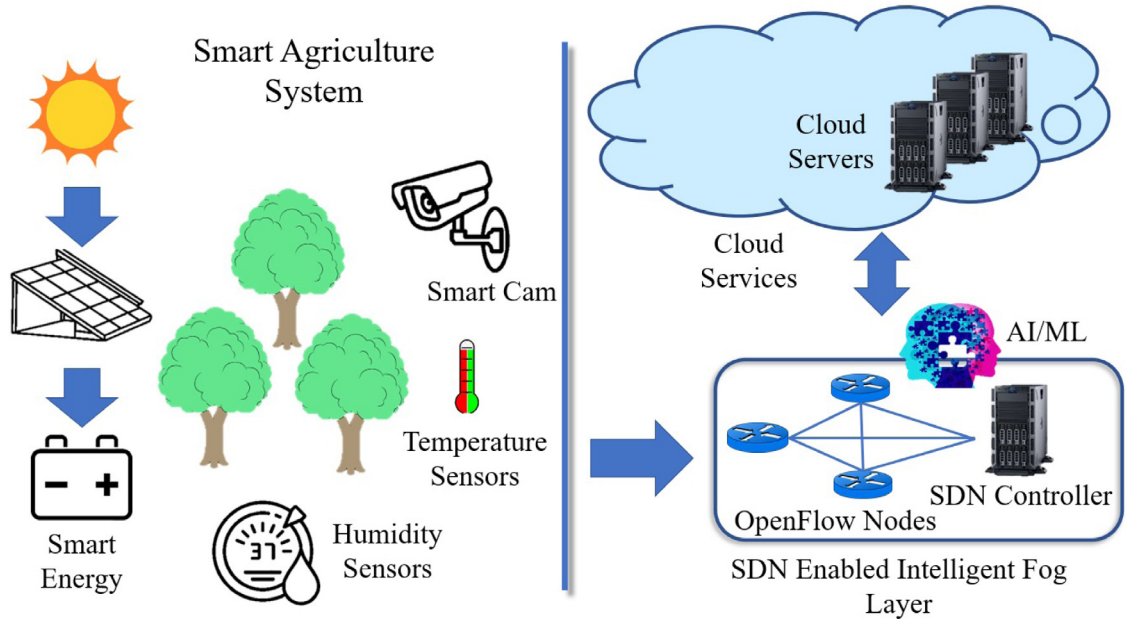


**Fig. 6.** Software-defined CPS (Smart Agriculture).

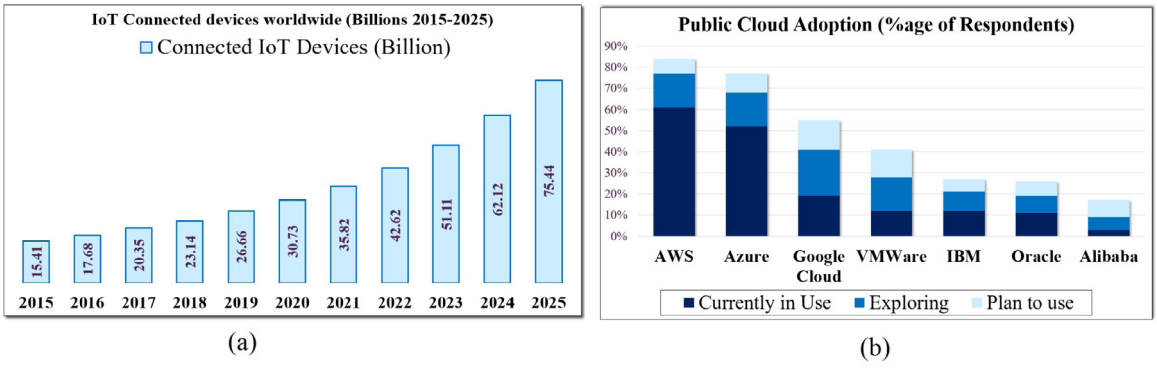


**Fig. 7.** (a) IoT Connected Devices Trend (b) Cloud Adoption Trend.

- **Huge amount of traffic data:** As more and more IoT devices are associating with a CPS system, there is an enormous amount of traffic data flowing between devices and the server. The CPS cyberspace server capability needs special attention while handling a burst of real-time data from geographically distributed heterogeneous devices.
- **Performance:** Performance talks about the processing speed which in turn takes throughput and latency into account [31]. The method of centralized decision making by the SDN controller for any new incoming traffic data in SDN brings the programmability but at the same time, it leads to performance issues.
- **Connectivity and Availability:** Connectivity refers to the connection between cyberspace and physical entities. Any disruption in the connection may affect the real-time data transfer to cyberspace. Similarly, availability refers to the time duration for which the SDN controller is functioning. In the SDN environment, controller, which is the ultimate decision authority, downtime leads to the unavailability of the whole network.
- **Scalability:** Scalability of the network is the network expansion capability with minimal efforts. In distributed or peer-to-peer controller architecture, westbound or eastbound APIs are required for communication between controllers.
- **Security:** Security refers to the jeopardizing the CPS components which may lead to loss/stealing of data, affecting the system performance and/or making the system unavailable for serving the intended operations.

When it comes to security, a significant effort has been invested by researchers to define data security [82]. The CIA triad is one such direction and well-known model that provides the fair universal mapping of data security issues to the vertices of the triangle namely Confidentiality, Integrity, and Availability (CIA) [83,84]. The CIA Triad defines the universally agreed-upon definition of data security for security professionals and policy-makers. The security breaches fall into one or more categories defined by the triad [85].

There are numerous vulnerable points in Software-defined CPS such as physical devices, network components, and cloud/cyberspace which can be mapped to all/any of CIA Triad vertices. The vulnerable CPS points are:

(i) **Physical devices**, the heart of the CPS system, sending the data to cyberspace and are outside cloud network and hence can be compromised easily.
(ii) **Software-defined network devices and controller**: The decoupling of vertical planes in SDN has made the amplification of attack surface. SDN suffers from a number

of security threats including but not limited to unauthorized access to the controller, forged traffic flows, lack of protocols to establish trust between the controller and third-party applications accessing the northbound APIs of the controller, and the well-known DDoS attacks that have an overwhelming impact on the network backbone.

(iii) **Cloud Security**: In CPS, most of the applications are hosted on the cloud which is vulnerable to attacks due to cloud architecture, and DDoS is one of the creepiest participants.

### 2.7. Security threats in software-defined cyber–physical systems

Though the software-defined cyber–physical system has proven a boon to modern society, it has opened the doors for vulnerabilities for several security issues. Almost every physical system is leveraging the use of sensors and actuators, it is hard to differentiate the legitimate user and malicious attacker because the physical devices are outside the firewall of cyberspace and can be compromised easily.

Moreover, SDN has helped in enhancing the security of the network by developing the traffic monitoring, analysis, and response extensions because of the extensible architecture of SDN. In a way, SDN supports the integration of security enhancement measures. The twin SDN technology, Network Function Virtualization (NFV) uses the virtualization of network functions and provides easy integration with the SDN controller. The security policies on the SDN controller can be added/updated and enforced in the network based on the traffic data analysis [81, 86]. Therefore, SDN offers a number of SDN-Supported security solutions.

Though SDN holds the key in mitigating the security breach in the CPS system, SDN security itself (SDN-Self) is still an area to be addressed. Implemented on Network Operating System machine (NOS) like any other application server and being the 'brain' of the network, stability, and security of controller is the prime concern for the network security enforcers [31,81,86].

There are a number of components widespread in CPS architecture. Hence, there are a number of security threats and challenges in Software-defined CPS ranging from perception layer (with IoT devices and sensors) to the Cloud platform and the underlying network infrastructure. Some of the attacks on various layers of SD-CPS are explained as follows:

#### 2.7.1. Perception layer attacks

(a) *Hardware Trojan Attacks:* Trojan is to compromise the IoT hardware to gain access to the software application running on the device or sensitive data present on the device. The attackers alter the hardware circuit and deploy the triggering mechanism of the Trojan to activate the wicked behavior of the Trojan [87].

(b) *Replication Attacks:* In replication attacks, any existing node can be replicated by malicious attackers. The replicated node imitates as an authorized node to the IoT device to gain access to IoT device data. Moreover, the replicated node can gain access to security credentials like certificates or encryption keys in the network or can revoke access to authorized nodes in the network [88].

#### 2.7.2. Cyber layer attacks

(a) *Identity and Access Management (IAM) Attacks:* Though cloud services are kept up-to-date keeping in view the spectrum of threats, frail identity, and nefarious usage of cloud platform have placed the cloud onto a significant position in the list of security threats [26].

(b) *Distributed Denial of Service Attacks:* The cloud availability has been a serious concern for the stakeholders as it may result in whopping financial losses [89].

**Table 2**
Taxonomy of communication layer (SDN) attacks in Software-defined Cyber–Physical system.

| Security issues | Affected network layers in Software-defined CPS | | | | |
|---|---|---|---|---|---|
| | AL | AC-I | CL | CD-I | DL |
| Unauthorized access | | | | | |
| Controller Access | N | N | Y | Y | Y |
| App Access | Y | Y | Y | N | N |
| Data leakage | | | | | |
| FR Discovery | N | N | N | N | Y |
| FP Discovery | N | N | N | N | Y |
| Data modification | | | | | |
| FR Modification | N | N | Y | Y | Y |
| Malicious applications | | | | | |
| Controller Commandeering | N | N | Y | Y | Y |
| Fake Rules Insertion | Y | Y | Y | N | N |
| Denial-of-Service | | | | | |
| C-S Command Flood | N | N | Y | Y | Y |
| S-F Table Flooding | N | N | N | N | Y |

Acronym Details: AL: SD-CPS: Software-defined Cyber–Physical System, Application Layer, AC-I: Application-Control Layer Interface, CL: Control Layer, CD-I: Control-Data Layer Interface, DL: Data Layer, App: Application, FR: Flow-rule, FP: Forward Policy, C-S: Controller-Switch, S-F: Switch-Flow.

#### 2.7.3. Communication layer attacks

(a) *Distributed Denial of Service Attacks:* The network is flooded and brought down by targeting the DDoS attacks on Network/Transport layer, Infrastructure layer attacks, and Application layer [90].

(b) *Eavesdropping Attacks (a.k.a. Data Sniffing):* Eavesdropping is listening to unencrypted data transfer in a network with the aim of obtaining sensitive information.

(c) Unauthorized access to the SDN controller.

(d) Unauthenticated access to SDN applications.

(e) Data leakage and data modification in form of flow rule or forward policy discovery and modification.

(f) Malicious Application such as taking over the controller commands (commandeering) or fake rule insertion.

Table 2 depicts the mapping of security threats and associated network layers in Software-defined Cyber–Physical System [91, 92].

Amongst the discussed security threats, DDoS attacks are the most notorious attacks which can degrade the network performance or make the CPS unavailable by disrupting the infrastructure layer or the cyberspace.

## 3. Fog computing — the extended cloud

Fog computing refers to the natural phenomenon of fog that exists between cloud and ground. The Fog Computing term was tossed by Cisco in 2012 as an extension to cloud computing [93]. Fog computing was introduced with the aim of overcoming the shortcomings of cloud computing and bringing the cloud services nearer to the physical world such as sensors, mobile phones, embedded systems, etc. Therefore, fog computing brings the computing, communication, and control elements closer to the 'Thing' devices. As compared to cloud computing, Fog stands in multiple dimensions [94,95].

- **Storage Capacity:** Fog nodes carry a substantial amount of storage near to the end-users rather than in remote data centers.
- **Capability:** Fog Computing, because of its vicinity to end-users, supports an assortment of application areas.

**Table 3**
Comparative view of Fog and cloud.

| Characteristics | Fog | Cloud |
|---|---|---|
| Location | Distributed | Centralized |
| Deployment | Small | Large |
| Operations | Less complex | Complex |
| Flexibility | Low specification machines | Highly computational machines |
| Internet connectivity | Intermittent | Always |

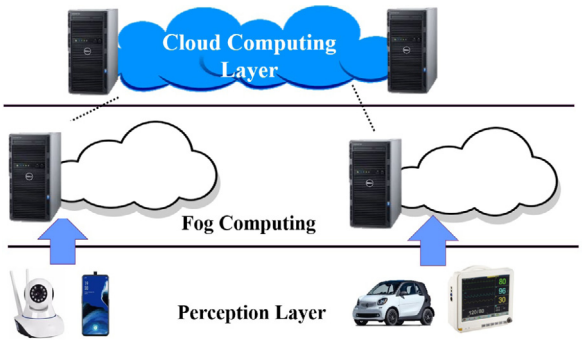


**Fig. 8.** Fog-Computing Eco-System.

- **Computation and Control:** Fog nodes are responsible for considerable control and computation near to the end-users rather than in remote data centers.
- **Communication and Networking:** Fog nodes carry out a significant communication and network function near to the end-users rather than diverting the traffic data to the cloud backbone network which makes the cloud network focus on its primary functions. Fog computing offers cloud-like services to network edge [96].

Hence, (a) Fog offers a wide range of services to cloud, (b) Fog is dependent on the cloud and vice-versa, (c) Fog and cloud can co-exists in a mutual beneficial manner.

The analogy of Fog Computing and Cloud Computing based on key parameters presented by Cisco is delineated in Table 3:

There is substantial debate on the roles and responsibilities of "Fog Computing" and "Edge Computing". Some of the researchers and academia folks believe in the true overlapping of fog and edge technologies. Although both technologies have the purpose to introduce a computation layer between perception and cloud layer, it is the way of data processing and location of processing the data that makes them different. The computation and storage are performed in a distributed architecture, between source and cloud, with fog nodes in the vicinity of cloud [97]. Whereas, edge computing pushes the computation facility near to data sources [98]. The computations are performed locally, however, the onus of data processing in distributed fashion remains with the fog nodes. Moreover, edge nodes do not offer cloud services like PaaS, IaaS, or SaaS. Fog computing is aimed to provide cloud services near to end-users.

Fig. 8 represents a pictorial view of Fog Computing where the 'Thing' devices send data to distributed Fog nodes and Fog nodes after applying the necessary computation, communicates the resultant information to cloud.

Fog architecture allows Fog nodes to distribute the control, computation, and storage services near to end-users. It has been proven using the information theory that a distributed system with delegated computational and storage complexities to the nearest point of presence (PoPs) outperforms the traditional defense system which consumes the victim-end resources for defense [99].

Fog computing helps in overcoming CPS challenges [94,100] as elucidated below:

(1) **Latency:**
Data analytics, control, computation and storage functions are performed near to end users thereby, enabling the CPS to meet time stringent requirements.
(2) **Network bandwidth conversation:**
For some of the CPS, it is not possible to transfer a huge volume of data to the Cloud. E.g. An A350 Jet produces 2.5 TB of data every day [101]. Nor, it is necessary to send the whole bunch of data to the cloud in most of cases. Thus, Fog offers the data processing near to the devices and send the computational results to Cloud.
(3) **Addresses Security concerns:**
Fog system can: (1) offer a range of security solutions, (2) time-to-time monitoring the device security status.
(4) **Services to resource constraint devices:**
Fog can offer resource-intensive services to resource constraint 'Things'.
(5) **Uninterrupted Service:**
A Fog node can operate autonomously thereby offering the uninterrupted services to the end-users.

Fog Computing is the latest trend and a revolution in network community that works hand-in-hand with Cloud computing, thereby, addressing the latency issues in the Cloud environment. Software-defined networking has proven a boon to trending technologies like IoT and CPS. Most of the tech giants, such as Google [44,102], Amazon [102] etc., are moving towards software-defined networks. The enterprises are adapting the trio — Fog Computing, SDN, and IoT.

## 4. Distributed Denial Of Service Attacks (DDoS/IoT-DDoS)

Distributed Denial of Service (DDoS) attacks are the cyber-attacks in which perpetrator makes the machine or network unavailable to legitimate users by disrupting the services of the victim server by sending network traffic flood from geographically distributed devices. In the technology breakthrough, the DDoS attacks are also targeted using prodigious IoT devices. The attacks can further be classified into:

### *4.1. Traditional Distributed Denial of Service Attacks*

The traditional DDoS attacks are targeted by compromising a massive number of computer devices distributed across geography which act as bots for launching the attacks. There is exhaustive research concluded so far on DDoS. The literature provides the details of DDoS Attacks Anatomy [103], DDoS modus operandi [104], various types of DDoS attacks based on a given criteria such as Network/Transport Layer DDoS attacks [105], SYN Flood attacks [90,106,107], ICMP Flood attacks [107,108], Application Layer DDoS attacks [109,110], HTTP Flood attacks [106, 111,112], Flash Events [110,113], SIP Flood Attacks [114,115], and Infrastructure Attacks [106,116]. Furthermore, Kaur et al. [117] have presented the taxonomy of Distributed Denial of Service attack tools.

The latest statistics, as reported by Neustar, shows the comparative details of DDoS attacks for Q3FY20 and Q3FY19 on Year on Year (YOY) basis [118] (In Fig. 9). The reports conclude growth in the number of attacks since the last year.

According to "Worldwide Infrastructure Report 2018" [119], a series of serious threats were reported in 2018 (viz. WannaCry, BadRabbit, etc.), DDoS has been reported as a topmost threat affecting 39% of enterprise organizations and is cited as topmost concern (with a predictable impact of 37% on enterprises)

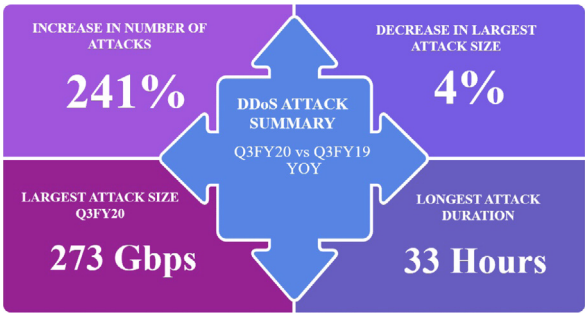


**Fig. 9.** Comparative Analysis (YoY) of DDoS — Q3FY20 and Q3FY19.

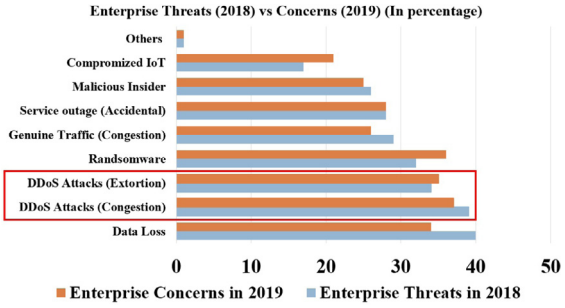


**Fig. 10.** Enterprise Threats (2018) vs Concerns (2019).

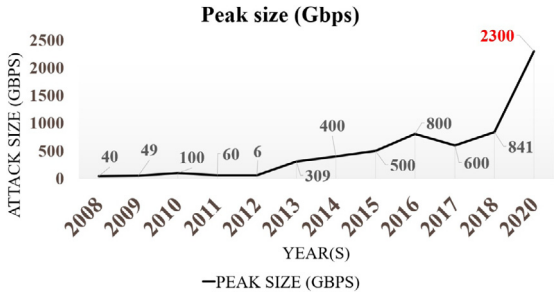


**Fig. 11.** Year-wise Peak Attack Size.

(Fig. 10) for 2019. The largest reported DDoS attack was 2.3 Tbps in 2020 (Fig. 11) which is the highest amongst reported in the last 10 years.

Also, DDoS attacks are the major methods to impact the physical world as well as cyberspace availability.

### 4.2. IoT-based Distributed Denial of Service Attacks

Safeguarding the server infrastructure and services offered by the cyberspace against highly devastating DDoS attacks is very difficult and a demanding need for today. In traditional systems, DDoS attacks are launched by an army of infected computers to exhaust the server resources. With the popularization of CPS and IoT based systems, new and more damaging DDoS attacks i.e. IoT-based DDoS attacks (As shown in Fig. 12) are into high attention which are launched by compromising the IoT devices. The results are devastating because the attack traffic is generated from millions of compromised heterogeneous IoT devices making the attack surface wider. Based on industry leaders (Arbor Networks [120], Akamai [121], Imperva [122] and Neustar [123] etc.) reports, the foremost attack paradigm affecting the service availability is shifting towards IoT-DDoS because of its scale, high complexity and frequency.

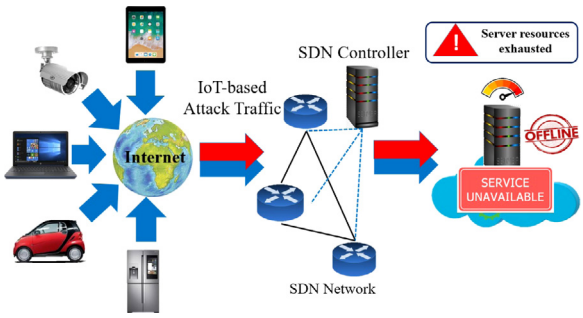


**Fig. 12.** IoT-based Distributed Denial of Service Attacks.

Examples include KrebsOnSecurity with an intensity of 665 Gbps [124], Dyn by Mirai botnet [125]. These types are categorized as IoT-based DDoS attacks and are impossible to trace back because of the wide adoption of IoT devices distributed across geographical areas. The massive amount of data traffic generated by IoT DDoS attacks directly affects the cost associated with developing and deploying the mitigation frameworks that only a few of the enterprises can afford to develop or deploy. Few of the examples of such enterprises are Google [126], CloudFlare [127] or Akamai [128].

In September 2016, a very high-intensity attack (of the order of 1 Tbps) was targeted on OVH (A French hosting provider) by Mirai botnet which took the business down for several hours [129]. Mirai botnet spreads by continuously scanning the Linux based IoT devices and exploits the vulnerability of such devices. Once compromised, these devices are injected with malicious programs turning them into bots.

With the popularization of CPS (having IoT, SDN, and Cloud Computing as key ingredients), the security aspects of CPS are becoming the major challenges. As per the latest report of Economic Times (Aug 2019), a growth of 22% has been reported in cyberattacks on India's IoT deployments during Q2FY20. With the reported statistics, India stood at the top position in the cyber-attack victim list with Mumbai, Bengaluru, and New Delhi attracting the maximum number of cyber-attacks [130]. Therefore, IoT-DDoS being the most devastating of the available evils needs special attention.

Hewlett-Packard (HP) in a press release reported the result of the study revealing that 70 percent of IoT devices are vulnerable to attacks [131]. However, very limited literature or solutions are available for IoT-DDoS attacks in Software-defined CPS. There is a strong need for research work for addressing the detection and mitigation of IoT-DDoS attacks in state-of-the-art Software-defined Cyber–Physical Systems.

The IoT-DDoS attacks are hard to handle because of very high traffic volume data, of the order of TB, generated by IoT devices [125,129]. The network hardware lying near to the end-users is not capable of performing complex calculations to learn the attack traffic and block the attack data because the operation is highly resourced intensive and require very high processing power. Though Cloud computing has great processing power, transferring the complete data stream to Cloud is not a feasible solution. Transferring such huge traffic to Cloud will increase the network bandwidth (may lead to exhausting network bandwidth). Moreover, DDoS attack detection methods implemented over Cloud will receive the legitimate as well as attack traffic, analyze traffic and send results back to the network enforcer to block the attack traffic will introduce latency in user operations. The concept of Fog computing was coined to conserve the network bandwidth, to reduce latency in network operations (and many more benefits). The mitigation frameworks leverage the

**Table 4**
Recent DDoS Incidents.

| Year/Month | Impacted body | Description |
|---|---|---|
| DDoS Attacks — Attack Category Unclassified | | |
| 2020 (Mar) | The U.S. Health and Human Services Department | U.S Health Agency faced DDoS attack amid COVID-19 outbreak [132]. Though the attack lasted for several hours, it could not bring any significant slow-down to agency's systems. |
| 2020 (Feb) | Amazon Web Services | AWS was hot hard with a record DDoS attack of 2.3 Tbps. However, the attempt was unsuccessful, the attempt continued for three days in February [133] |
| IoT-based DDoS Attacks | | |
| 2019 (Mar) | University of Albany, USA | Seventeen attacks were targeted which brought the University server down for 5 min [134]. |
| 2018 (Mar) | Wired Telecommunication Carriers | USA based "Wired Telecommunication Carrier" faced massive 1.7 Gbps mem-cached attack which was mitigated by NETSCOUT Arbor solutions [135]. |
| 2016 (Sep) | French hosting provider (OVH) | The French hosting provider was hit by Mirai botnet attack with an intensity close to 1 Tbps [136]. |
| 2016 (Oct) | Dyn | Dyn was targeted by Mirai botnet with a traffic impact of 1.2 Tbps resulting in unavailability of more than 85 popular websites including Netflix, Paypal, Amazon with a loss of $110 million [136]. |
| Traditional DDoS Attacks | | |
| 2019 (Oct) | Organizations in Scandinavia and Southern Europe | High thump TCP SYN reflection and amplification attacks by leveraging the carpet-bombing DDoS attack techniques [137]. |
| 2019 (Feb) | National Union of Journalists of the Philippine | The site was hit with an impact of 468GB/s of traffic and with a downtime of several hours [134]. |
| 2018 (Feb) | GitHub | GitHub, a popular developer platform, faced sudden assault of traffic at record breaking 1.3 Tbps per second rate [138]. |
| 2017 (Dec) | Bitfinex exchange | Bit Gold site (New currency introduced by Bit Coin) under DDoS fire as a result Bitfinex was hit hard and cybercriminals attempted to manipulate the currency rates [139]. |
| 2017 (Jan) | Saudi Arabia Sadara Chemical Company | DDoS attacks were launched to express the political protest in Saudi Arabia. [140]. |
| 2017 (Jan) | Ukrainian Power-grids | DDoS Attack was launched due to political reasons [140]. |
| 2015 (Feb) | UAE energy sector | The attack was launched as a result of political activity. [140]. |

concept of Fog computing for early detection of IoT-DDoS attacks as well as to provide better response time as explained in the next section.

### *4.3. Role of SDN (in contrast to traditional networks) in DDoS defense*

The Software-defined networks help in providing resilient architecture for defending DDoS attacks. The open-flow routers can be remotely configured, SDN controllers can be load balanced. Furthermore, DDoS detection and mitigation solutions can easily be deployed onto cyber–physical systems because of SDN and network function virtualization (NFV) integration. The artificially intelligent algorithms (deep learning and machine learning) can easily be deployed and integrated with SDN. The aforesaid architecture and capabilities are not possible using traditional networks. Hence, SDN provides the dominion to the cyber–physical system.

### *4.4. Recent DDoS incidents*

The Table 4 show recent DDoS attacks, impacted bodies, and description of the event. The attacks are classified into IoT-based DDoS Attacks, Traditional DDoS Attacks, and the DDoS Attacks where the impacted bodies have not disclosed the DDoS attack category. The mentioned attacks are either on the application layer or the network layer.

## 5. Literature survey

In this section, several noteworthy works and investigations carried out to address DDoS as well as the trending IoT-DDoS-based attacks have been discussed. The majority of the literature survey will cover the study of DDoS (IoT-DDoS and DDoS) attack detection and/or mitigation frameworks in SDN enabled networks (SDN-Supported) including but not limited to SDN enabled Clouds, SDN Enabled simple networks, Fog Assisted systems and Software-defined CPS system. Due to the architectural design perspective, SDN itself is vulnerable to attacks [81]. SDN has a big picture of the underlying network and possesses the capability to modify the underneath network that makes it vulnerable to threats. Attack detection and mitigation against the SDN environment (SDN-Self) will also be reviewed. Table 5 presents the organization scheme for the literature survey and maps the literature survey sections to designated categories.

**Table 5**
Traceability of literature survey.

| | Fog | | Non-Fog | |
|---|---|---|---|---|
| | DDoS | IoT-DDoS | DDoS | IoT-DDoS |
| SDN | A | B | C | D |
| Non-SDN | G | H | E | F |

The traceability of literature survey maps the infrastructure configuration of the proposed solutions as follows:

(A) **Software-defined Networks and Fog Computing based solutions against DDoS Attacks**
Priyadarshini et al. [96] have proposed an intelligent framework with a blend of deep learning algorithms and fog computing paradigm to mitigate DDoS attacks. The authors have proposed a novel source-based defender. In the approach, the SDN controller hosts the defender to defend against DDoS attacks targeted at Network/Transport layer. In contrast to several solutions with shallow-learning approaches, the authors have come up with a variant of the deep learning algorithm, Long Short-Term Memory (LSTM). They have chosen the LSTM model because of its best suitability with sequential and time-collected data. The Deep Learning model is built by using the Python library 'Keras'. Authors have further improvised the LSTM model to avoid

overfitting and vanishing gradient problems. The improvisation is achieved by integrating dropout probability and Mini Batch algorithms. The authors have tested the solution on ISCX 2012 and CTU-13 Botnet datasets as well as the dataset generated in the simulated environment. The DDoS attack traffic was generated using an open-source utility (HPing-3). The experimental testbed is configured in a layered architecture comprising of Cloud, Fog, and Application layers. The Cloud is setup on 'ownCloud' with a number of enabling open source technologies viz. Apache Web Server, PHP, MariaDB (A Fork of MySQL). FloodLight is used as an SDN controller and network topology for multiple virtual machines at Application layers is created using Mininet emulator. The algorithm was validated with 90:10 training and test set proportions and a 10-cross validation scheme. The authors were able to achieve an accuracy of 98.88% with a dropout rate of 0.2 for all hidden neural network layers.

In a proposal for Medical Cyber System (MCS) which is Fog Assisted and leverages IoT technology for cardiovascular disease affected patients, authors, Karthick et al. [141], have proposed a Fog layer to de-centralize the computing infrastructure which includes data processing, storage, and security. The described the Fog to be an effective place between data source and cloud. They have proposed to enforce a novel policy-based access mechanism in the Fog layer. They also suggested a virtualization feature in the edge network which will help to create a flexible and manageable policy-based network environment by leveraging the SDN networks and Network Function Virtualization (NFV). The whole system is divided into a Master policy server, which is a policy repository and rules pertaining to IoTs and Fog policy managers to enforce the policies in the network. The proposed approach was tested with real-time data of cardiovascular affected patients using an IoT kit. The authors were able to infer that the early decision taken by the Fog layer rather than Cloud being the computational point has increased the system performance.

(B) **Software-defined Networks and Fog Computing based solutions against IoT-DDoS Attacks**

In an attempt to design a DDoS Mitigation Framework which includes multiple defense layers i.e. Perception Layer at the edge of the network, Cloud Computing Layer at the top of the hierarchy and Fog Computing Layer in between Fog and Cloud for Industrial Internet of Things (IIoT), Yan et al. [142] have proposed an abstract design of framework architecture. The framework is proposed with multiple layers in a vertical plane for DDoS detection and mitigation. An Edge layer is proposed near to the end user IIoT devices and a Fog layer is suggested to reside between Edge and Cloud layer. The IIoT devices are said to be at the Perception layer. Therefore, Perception, Edge, Fog, and Cloud are the multilayer backbone of the framework. In the paper, the authors proposed the responsibilities of each layer where the edge layer is expected to handle IIoT connections, perform firmware security checks, access control, malicious software detection, intrusion detection, attack reduction, Honeypot monitoring, and communication data encryption. While the Fog layer is responsible for traffic data collection, DDoS detection, suppressing the DDoS attacks, status perception, and IoT service. Though, most of the work is performed by Edge and Fog layer, Cloud layer is also suggested to have additional responsibilities of DDoS detection by means of Big data analysis using Big data technologies like Apache Hadoop, Spark, etc. Also, a suggestion to use the neural networks and deep learning is made at the Cloud computing level do perform the intelligent processing involving a high level of computation. The authors presented the experimentation results for TCP SYN flooding attacks and Ping-of-Death attacks. The experimentation facility involved Intel i5 4570 3.2 GHz processor with 8 GB RAM. Mininet is used as a network simulator. The authors have used 'Snort' as IDS system. In the presented paper, authors have not discussed or proposed any of the novel algorithms to detect or mitigate the DDoS attacks. By means of experimentation, authors were able to show the performance improvement in the system by 37.03 percent. In [143], the authors, Bhardwaj et al. of IoT-DDoS prevention framework based on edge computing technology have described that IoT-DDoS attacks are launched using infected IoT devices in a cyber–physical system and are one of the emerging security issues. Application layer with the sheer amount of data being transferred from the IoT devices, traffic generation from distributed locations, and seemingly legitimate nature of traffic makes the mitigation multifaceted. They have implemented the solution at the edge because the edge has the capability (Although limited capability) for IoT traffic packets processing. They have proposed an edge architecture to serve as an initial point of defense and named it ShadowNet. The edge locations have been set up to profile the stream of IoT packets at the given edge location and then transfer the IoT traffic information, shadow-packets, to ShadowNet service. ShadowNet service is configured to detect imminent DDoS generated by IoT devices and mitigate with defense action. The authors have prototyped ShadowNet edge functions and services in the Go programming language. For experimentation setup, the GENI platform with four virtual machines at different locations interconnected by fast ethernet and underneath SDN networks have been used. The systems (VMs) under testbed are equipped with Intel Xeon Processors 2.67 GHz core and 1 GB RAM each. Attack traffic is generated with HTTP GET flooding and UDP flooding characteristics to simulate the traffic data from all types of sensors and video surveillance cameras. For HTTP GET flooding BoNeSi is used which generates the request at 500 Req/s rate. The authors, in their experiment, were able to demonstrate that ShadowNet was able to prevent 40% to 82% of damage from HTTP and UDP flooding respectively and it was able to detect the attack in 0.62 s.

(C) **Software-defined Networks based (Fog Computing absent) solutions against DDoS Attacks**

In [144], Bawany et al. have proposed a DDoS mitigation framework for SDN enabled cyber–physical system. They have used smart cities model to implement the solution. The authors have termed the mitigation framework as the Secure and Agile framework because it covers the wide range of applications in smart cities, therefore, adaptive in nature. The authors have proposed three traffic filters i.e. Proactive, Active, and Passive filters to satisfy the varying needs of a wide range of applications. Proactive filters have been proposed for critical applications such as Smart Traffic Control, Smart Grid, Smart Emergency services, etc. Active and Passive filters have been designed to cover the applications which have moderate to soft security constraints. Location based services and Weather applications have been defined to have moderate to soft security requirements. The authors have developed the framework in stages viz. defining a threat model, outlining the smart cities security requirements, proposing the framework architecture based on threat model, proposing the defense modules at various framework levels (viz. Data plane, control plane, and application plane) and implementing the

load balancing algorithms to satisfy the performance and scalability requirements. The experiment has been performed with Mininet hosted on the Ubuntu machine with Intel Core i7 (3.67 GHz) and 8GB memory. Iperf has been used to generate the DDoS traffic stream. The legitimate traffic is generated at real-time traffic rate (30 Mbps: TCP-85%, UDP-12%, and ICMP-3%). The attack traffic ranges from 20 Mbps to 80 Mbps. The authors were able to show by means of experimental results that false negatives were reduced to 0% for critical applications using Proactive filters. The accuracy lies in the range of 30%–50%. Through Active filters, they achieved the accuracy of 90% with False positive (FP) as 7% and high False Negatives (FN) as 25%.

A multi-vector DDoS detection model which is based on Deep Learning techniques has been proposed by Niyaz et al. [145]. They worked on detecting various spoofed attacks including TCP, UDP, and ICMP. The detection system is implemented as an application by leveraging the northbound interface of the SDN controller. The implementation contains a traffic collector for TCP, UDP and ICMP flow and then the detection system triggers the Feature Extractor module. For experimentation facility, real network traffic in the home wireless network (HWN) and a private testbed with POX controller is used. The DDoS attack is generated using hping3 utility. They had developed Sparse Auto-Encoder (SAE) with Recursive Neural Networks (RNN). A separate detection model using soft-max and the neural network had been developed by the authors. They used Intel Core (i7 - 3.40 GHz processor) with a 16GB RAM configuration machine. MATLAB 2016a has been used for algorithm development purposes. The training time (81,010 records) and classification time (34,717 records) noted was 524s and 0.0835s respectively. By means of experimental results, they showcased that the system has an accuracy of 95.65% for the 8-class classification model and 99.82% for a 2-class model with False Positive rate of 0.5% and 0.3% respectively. However, the authors informed the system limitation in terms of processing capabilities and suggested using distributed computing for better performance of algorithm execution.

In [146], Buragohain et al. in their experimentation and proposal for DDoS attack detection and mitigation have come up with FlowTrApp design which can be used in SDN based data centers to address DDoS issues. The app detects and mitigates using the bound checks on flow rate and flow duration. These parameters are the decision-makers if traffic is being sent by legitimate users or not. They described that TCP, UDP, and ICMP flooding attacks send a huge number of fake packets to victims to exhaust the victim resources. On the other hand, L7 attacks are much more complex attacks to be detected. The authors have classified the traffic into different categories based on flow rate and flow duration in order to detect and mitigate the DDoS attacks. They set the session limit per IP address for HTTP traffic. They have tested the proposed solution on the simulation environment with Mininet as a network emulator, Floodlight controller, and sFlow as traffic collector. The network topology used as Fattree topology. In the experimentation setup, OpenFlow enabled switches have been incorporated. The overall performance of FlowTrApp has been compared with Quality-of-Service (QoS) based mechanism and load balancing scheme. FlowTrApp performed better in both cases with a reduction of load on the controller.

(D) **Software-defined Networks based (Fog Computing absent) solutions against IoT-DDoS Attacks**

Nam et al. [147] have proposed SDN based DDoS detection approach leveraging a blend of self-organization maps (SOM) and K-Nearest Neighbors technique. They have implemented the proposed detection approach in SDN supported environment. They have implemented the solution in the form of four sequential modules i.e. monitoring, detecting, alerting for the attack, and offering the mitigation against DDoS attacks. The authors have used the "DDoS Attack 2007" dataset and "CAIDA" dataset for experimental study. The authors have used entropy to measure the degree of divergence due to randomness in the extracted features. They have used total packets and entropies of source address, source port, destination port, packet protocols for analysis and classification. The experimental setup included a POX controller on a system with the Intel Core i5 processor with 4 GB RAM. TcpReplay was used to generate traffic from the dataset. They used different blends of k-NN and SOM. They were able to achieve 99.05 Detection Rate (DR), 2.74% of False Positive Rate (FPR) with-in 22.93 ms of processing time for the k-NN algorithm. Also, SOM and k-NN combined together reported 98.24% DR, 2.14% of FPR in 2.8 ms of processing time. However, they further described that the traffic may be different for different types of devices which they are yet to automate in their future work.

Sharma et al. [148] leveraged distributed Blockchain (DBC) [149] technology to propose a defense solution for IoT devices using SDN technology. The proposed solution is low-cost, secure, and provides on-demand access to computing infrastructure. The paper also proposes secure, distributed SDN enabled cloud architecture and utilizes DBC to gather the traffic data, perform classification, and analyzes the IoT stream. The traffic data has been collected in real-time from the author's own private cloud and Amazon's EC2 Cloud data-center. The testbed in experimentation used Intel i7 machines with 64 GB RAM. TFN2K tool had been used to generate the ICMP, TCP, and UDP flooding attacks.

Ozcelik et al. [129] have come up with a defense solution against IoT-DDoS attacks using SDN networks. The defense solution exists at the edge of the network closer to IoT devices. A study on Mirai botnet has been taken in due consideration while proposing the solution. The Edge solution is proposed in order to use the SDN controller capabilities rather than integrating the whole solution at the cloud level and overburdening the cloud with communication and computation overheads. Authors reviewed that in traditional DDoS attack detection approaches, the detection and mitigation are made the responsibility of the target environment. The mentioned approach is a reactive approach. However, SDN controller capabilities could be used to uplift the detection and mitigation of a "preventive" approach. In the proposed Edge-Centric-Software-Defined IoT Defense (ECESID) framework, authors have proposed two algorithms namely Threshold Walk Credit Based Limit (TRWCB) and Rate Limit (RL) algorithms. The approach uses the scanning of phase and traffic from bots to see if any of the nodes have been compromised. The infected nodes are deleted and related rules are installed on the switches. The authors have used Mininet Wi-Fi to emulate the IoT devices and OpenFlow enabled Wireless Access Points. Floodlight has been used as the SDN controller and iperf as a traffic generation tool. All the simulations were executed on the Ubuntu machine with Quad-core 2.60 GHz. The authors have inferred from the experimentation results that to detect an infected node, it took 6.02 s of time with maximum

and fastest detection times were reported as 11.8 and 3.6 s respectively. Though for the application layer attacks, it is too early to provide a detection or mitigation technique at edge level. Here, authors have not given due consideration to Fog based defense mechanism which lies in between Edge and Cloud layers.

Bhunia et al. [150] in their paper cited the complexity of IoT-DDoS attacks because of high traffic volume and heterogeneity. They have proposed an SDN-based secure IoT framework known as SoftThings. SoftThings detects abnormal traffic behavior. They have evaluated linear and non-linear versions of Support Vector Machines (SVM) for attack traffic detection (Specifically TCP flooding). Also, the experimental study included attacks on IoT devices as well as from IoT devices. Authors were able to achieve 94% precision and 92% recall for linear SVM whereas non-linear SVM reported 98% precision and 97% recall for the attacks targeted to IoT devices. The second experimentation facility (IoT devices as attackers and attack type as ICMP flood) reported 92% and 97% precision for linear and non-linear SVM respectively. The recall was recorded as 88% and 96% respectively. The case study infers improved results for a non-linear version of SVM. The experimental setup consisted of IoT devices, SDN-Enabled switches, Cluster SDN-Controller, and Master SDN-Controller. The authors categorized anomaly detected into the Learning Module and Classification module. Based on the Classification Module, they prepared appropriate rules dynamically and enforced the rules onto the switches. The testbed consisted of a Mininet emulator supporting OpenFlow switches. IoT devices were also emulated using Mininet. POX was used as an SDN controller where algorithms are executed for DDoS detection. TCP flooding, ICMP flooding were used as attack scenarios during the experiment.

(E) **Fog Computing based (Software-defined Networks absent) solutions against DDoS Attacks**

Cloud computing has gained a heap of popularity because of its uniquely scalable, elastic, on-demand, and pay-as-you-go service model. Attackers' phase shift towards the cloud environment is not a surprise. Deepali et al. [151] have proposed a DDoS defense framework for Cloud which uses computation nodes at the Fog computing layer. In the proposed framework, the traffic is made to pass through a defender at the fog computing layer. The experimentation setup consisted of several Linux and Windows machines. Windows machines were configured to generate legitimate traffic while the metasploit interface on Kali Linux was used to generate the SYN flooding attack and Ettercap was used to generate the DDoS attacks from spoofed IP addresses. Also, the LOIC (Low Orbit Ion Cannon) tool was used to generate the TCP, UDP, or HTTP traffic for attack purposes. The authors have used 'tshark' to capture the packets on the fog layer whereas the 'netstat' tool was used to obtain network statistics. Authors concluded from the analysis of attack traffic in contrast to legitimate traffic that attack traffic had a smaller traffic size. They generated the attack in three phases i.e. (a) SYN Flooding, (b) using Ettercap plugin for generating man-in-the-middle (MITM) attack and, (c) using LOIC tool to generate an attack. In each of the scenarios, they were able to drop the attack packets. However, high-level architectural details of the defender system are not mentioned. Also, authors have not discussed performance parameters of the fog defender system.

Paharia et al. [152] in their work have used Fog computing as a defensive approach against DDoS attacks. The proposed architecture obstructs the attack traffic generated by DDoS attacker by taking advantage of Fog computing. They described that Fog is used as a filtering layer between user and Cloud. The authors have overcome the reduction in traffic to Cloud and hence achieved improvement in network performance. In the proposed architecture, traffic filtering is done in different phases viz. IP verification and Captcha checking, the setting of IP addresses, and tools for analysis. The experimentation testbed used EtherApe, NetGrok, VizNet tools for network traffic monitoring. However, the framework lacks the methods for detection of DDoS in a CPS/IoT based environment where it is not a feasible solution to check the IP addresses, map them with the legitimate IP list, and provide a defense mechanism based on IP filtering and Captcha verification.

(F) **Fog Computing based (Software-defined Networks absent) solutions against IoT-DDoS Attacks**

Zhou et al. [153] have implemented a fog-based mitigation framework against DDoS attacks in Industrial-IoT (IIoT) systems. As IIoT devices are prone to DDoS attacks because of their limited computational ability and distribution across geography and weak security shields, the authors have proposed the real-time traffic monitoring using virtualized network function (VNF) on local servers. They featured three-layer architecture with layer-1 responsible for traffic filtering using firewalls which offers rule-based filtering, layer-2 performs specification-based analysis of network traffic by leveraging virtual network functions (VNF) on local server and layer-3 for co-ordination and consolidation of information from distributed fog nodes. The most widely used specification protocol Modbus/TCP was used for the experimental study. The framework was implemented on top of SCADA (acronym for Supervisory Control And Data Acquisition) testbed in industry. In the experimentation facility, the field devices consist of several Programmable Logic Controller i.e. PLC devices, analog and digital input/output devices like sensors, smart cameras, and traffic simulator programs. Fog layer consists of local servers which connect to Cloud. VNF runs on local servers to monitor traffic in real-time. The Fog processing servers used in the experimental testbed consists of DELL PRECISION T3500 PCs, 16 GB memory with Windows 7/Linux systems. The Ubuntu Iptable functionality has been used to set up rule-based filtering in firewalls. The authors have used 'Githuba', a GitHub project, together with snort to present the detection statistics in a web-page. For one of the scenarios, DDoS traffic (ICMP) is generated using Smurf. From the experimental study using the fog computing approach, it is evident that the detection time was 129 ms with an average detection rate of 99.84% for TCP and 145 ms for Modbus with an average detection rate of 88.02%.

Dorri et al. [154] have presented Blockchain (BC) based approaches to provide distributed security and privacy. They have taken Smart Home, a cyber–physical system, as their case study to apply the Blockchain-based solution to address the DDoS and other attacks. Though BC-based work involves significant amount of delays and computational overheads, it is not recommended for resource constraint IoT devices. The approach is classified into three tiers, namely: cloud storage, overlay, and smart home. The authors have used BC to control and audit communication. To add any of the devices to Smart home in BC monitored network, genesis transaction is performed and key sharing with the device is using a generalized Diffie–Hellman algorithm. The initialization and execution of the BC network are done using the Smart Home Miner which acts as Fog

node between the IoT devices and the Smart Home system. For multiple homes, separate miners and storage for each home are required. The authors explained that it would be impossible for the attackers to install malware into IoT devices in a BC managed smart home. Even if the attacker is able to compromise the IoT device, the authors have provided a second level of defense i.e. all the outgoing traffic is examined by a miner by checking the policy header. Though checking of CH key lists and changing the PK in CH list were outside the scope of the paper. The authors have simulated the Smart Home scenario in the Cooja simulator and used IPv6. From the experimental results, it is evident that BC methods increase energy consumption because of CPU intensive tasks, transmission, and listening.

(G) **Conventional solutions (Software-defined Networks and Fog Computing absent) against DDoS Attacks**
Ko et al. [155] have proposed a DDoS mitigation approach based on deep learning model. They proposed the solution that can be implemented within the ISP domain as the ISP domain is the junction in the network that connects users to the internet. They have proposed a stacked self-organizing map (SOM). The authors deployed 3-Layered SOM for algorithm implementation. The approach is based on an unsupervised deep learning algorithm. They have used Apache Spark for distributed big data analysis to lessen the gravity of the high-volume attack. The authors validated the solution to NetFlow data collected by the ISP. As Netflow data volume produced by the ISP is too high, they have employed methods to boost system performance and data size reduction. They have used Apache Spark for boosting the system performance as it offers fast distributed computing and enables scalability. For the later slice of a solution, i.e. data size reduction, they have implemented dynamic feature selection in SOM layers. They also focused on keeping the neuron numbers to small for increasing the processing speed. They have used the BoNeSi simulator tool from GitHub repository [156] for ICMP, UDP, and TCP SYN flood attacks. The attack traffic generator used 50,000 spoofed IP addresses for the attack. The MinMaxScalar was used for data normalization. The experimentation was performed in two stages, one with 78 normal IP addresses and another with 281 normal IP addresses. For earlier cases (78-Normal IP addresses), the performance parameters as noted during the study were: 100% Recall, 99.65% Precision, and 99.82% FI score for UDP flood. For ICMP flood, parameters noted were: 100% Recall, 98.73% Precision, and 99.36% FI score. Whereas, performance parameters for TCP SYN flood were: 100% Recall, 99.14% Precision, and 98.57% FI score. When the experiment was repeated on the second target with 281 normal IP addresses, performance parameters for UDP flood, ICMP flood, and TCP SYN flood were: (a) 100% Recall, 99.65% Precision, 99.82% FI score, (b) 100% Recall, 98.73% Precision, 99.36% FI score and, (c) 100% Recall, 99.76% Precision, 99.88% FI score respectively. As authors have used dynamic feature selection technique which is expensive in terms of computational resources. As a future direction, they plan to work on increasing dynamic feature selection efficiency.

(H) **Conventional solutions (Software-defined Networks and Fog Computing absent) against IoT-DDoS Attacks**
The Internet of Things (IoT) concept has enabled the interconnection of pervasive connected devices representing the Cyber–Physical link. The pandemic growth of distributed, heterogeneous devices has offered an easy platform to attackers to compromise the system availability by generating DDoS traffic from compromised IoT devices. Cvitic et al. [157] in their work have proposed a novel approach for IoT-DDoS traffic detection by classifying the IoT devices. The work is a continuation of their earlier work on IoT-traffic characterization for Smart Homes carried for DDoS detection [158]. The authors have taken a Smart Home case study, came up with differentiation between Machine Type Communication (MTC) and Human Type Communication (HTC) network traffic for their research work. They have identified that MTC traffic, as compared to HTC traffic, transmits a smaller number of packets, has long duty-cycle patterns, similar traffic characteristics for a single device, dependency of traffic on certain events, and traffic aggregation. From detection capability aspects, they also found that MTC traffic could also be classified as belonging to a group of devices. The authors have developed a DDoS detection conceptual model. The authors proposed an approach that is based on the class affiliation of IoT devices. The proposed four-phased model runs through a collection of network traffic, pre-processing the collected traffic data, feature selection, and determining the device class affiliation. They applied logistic regression to identify device class affiliation based on predictor feature of the traffic. The final classification phase involved, AdaBoost, a machine learning method. The boosting model aimed at deviation detection in IoT device characteristics which inferred that DDoS traffic was generated in the time frame under observation. As a future direction, authors have plans to analyze traffic data from more IoT devices.

The summarized view of the literature survey (presence or absence of the essential components of a quintessential cyber–physical system) is presented in Table 6.

## 6. Amazon web services (AWS) - a cloud case study for DDoS attacks mitigation

Amazon Web Services (AWS) [159–161] is one of the renowned information technology giants best known for cloud computing services. As cloud services are an inseparable component of the cyber–physical system, close attention has been paid to security service offered by AWS. AWS offers a set of services and flexible infrastructure for hosting and/or creating highly available applications. It gives the resiliency against DDoS attacks in no time. The specialized services like Amazon CloudFront, Elastic Load Balancing, AWS WAF [162], Amazon Route 53 [163], and Elastic Load Balancing [9] enables users control the network traffic and unsolicited requests are obstructed. The mentioned services are integrated with AWS Shield [164] to mitigate DDoS attacks. AWS has released best practices for DDoS mitigation as under:

(a) Scalable Infrastructure
(b) Decoupled architecture for application development. For example, decoupling the front end, media, database, static contents, etc. limits the access to critical application components, hence, focus on DDoS mitigation efforts to publicly available application components.
(c) Logging and monitoring of traffic and infrastructure in order to know the differences between benign and attack requests.
(d) Deploying resilience infrastructure.

AWS promises high availability of infrastructure (for IaaS) and applications (for SaaS), when developed/deployed in conjunction with AWS state-of-the-art web-services. These services are:

**Table 6**
Summary of the literature survey.

| Reference | Year | SDN | SDN-Type | Fog | IoT-DDoS | Defense (D/M) | Env S/RT | CPS | PM |
|---|---|---|---|---|---|---|---|---|---|
| Priyadarshini et al. [96] | 2019 | ✓ | SUPP | ✓ | ✗ | D, M | S | ✗ | ✓ |
| Ozcelik et al. [129] | 2017 | ✓ | SUPP | ✗ | ✓ | D, M | S | ✗ | ✓ |
| Karthick et al. [141] | 2019 | ✓ | SUPP | ✓ | ✗ | – | RT | ✓ | ✗ |
| Yan et al. [142] | 2018 | ✓ | BOTH | ✓ | ✓ | D, M | S | ✓ | ✓ |
| Bhardwaj et al. [143] | 2018 | ✓ | BOTH | ✓ | ✓ | D, M | S | ✗ | ✓ |
| Bawany et al. [144] | 2019 | ✓ | BOTH | ✗ | ✗ | D, M | S | ✓ | ✓ |
| Niyaz et al. [145] | 2017 | ✓ | SELF | ✗ | ✗ | D | RT | ✗ | ✓ |
| Buragohain et al. [146] | 2016 | ✓ | SUPP | ✗ | ✗ | D, M | S | ✗ | ✗ |
| Nam et al. [147] | 2018 | ✓ | SUPP | ✗ | ✓ | D | RT | ✗ | ✓ |
| Sharma et al. [148] | 2018 | ✓ | SUPP | ✗ | ✓ | – | RT | ✗ | ✗ |
| Bhunia et al. [150] | 2017 | ✓ | SUPP | ✗ | ✓ | D, M | S | ✗ | ✓ |
| Deepali et al. [151] | 2018 | ✗ | – | ✓ | ✗ | D, M | S | ✗ | ✗ |
| Paharia et al. [152] | 2018 | ✗ | – | ✓ | ✗ | D, M | S | ✗ | ✗ |
| Zhou et al. [153] | 2019 | ✗ | – | ✓ | ✓ | D, M | S | ✗ | ✓ |
| Dorri et al. [154] | 2017 | ✗ | – | ✓ | ✓ | D | S | ✓ | ✓ |
| Ko et al. [155] | 2020 | ✗ | – | ✗ | ✗ | D | S | ✗ | ✓ |
| Cvitic et al. [157] | 2019 | ✗ | – | ✗ | ✓ | D | S | ✓ | ✗ |

**Acronym Description:** SUPP: SDN-Supported Security Solution, SELF: Security solution for SDN using SDN, D: Defense, M: Mitigation, Env: Setup Environment, S: Simulated, RT: Real-time, PM: Performance Metric Discussed in paper?

(a) **AWS Shield:** AWS Shield is a DDoS protection service available in two flavors to the user community: Standard and Advanced. Standard service is implicitly available with elastic load balancers and other such services. However, advanced AWS Shield is an add-on al-a-carte service that enables the resource level monitoring of application-level traffic.
(b) **AWS WAF:** AWS WAF is an acronym AWS Web-Application Firewall that provides protection against common web-exploits effecting the application availability or excessive consumption of resources. It can be used in conjunction with AWS Shield. AWS WAF uses rules/policies to target specific conditions or requests like implementing size constraints to block web-requests, applying geo-restrictions for country-specific requests.
(c) **Amazon Route 53:** Amazon Route 53 is a highly available DNS service and is capable of managing the traffic globally.
(d) **Elastic Load Balancing:** The said service automatically distributes the incoming traffic across multiple infrastructure or software elements like EC2 (Amazon Elastic Cloud Computing) instances, availability zones, etc.

Amazon Web Services provides world-class infrastructure for application, platform, and infrastructure hosting and offers a number of services against DDoS mitigation. However, the services are too tightly coupled with Amazon accounts and are available as al-a-carte software components rather than services that these services cannot be leveraged in standalone mode by the applications hosted on third-party accounts. Moreover, the services are available in a closed system that restricts the attack data unavailable for the research community.

## 7. Retrospection and vulnerability analysis of solution space from the literature

The core of the retrospection lies in *'What went well'* and *'Scope of improvement'*. The first part of the section discusses the best practices derived from the solution space. The later of the section focuses on vulnerability study, open research issues, and challenges.

### 7.1. Retrospection — what went well

The solution space, discussed in the literature, consists of the state-of-the-art novel solutions that are proposed against DDoS and IoT-DDoS attacks detection and mitigation. The excerpts from the literature survey which is the need of the hour for an effective defense solution against DDoS/IoT-based DDoS are collated in Table 7.

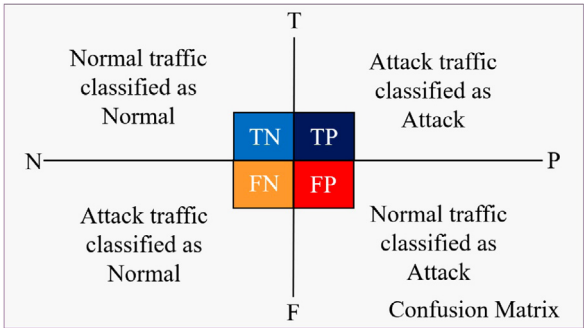


**Fig. 13.** Confusion Matrix for Performance Evaluation.

### 7.2. Retrospection — scope of improvement

The research gaps from the literature study are summarized as follows:

#### 7.2.1. Lack of discussion on every single performance measure

Although there are a number of solutions available in SDN, IoT, DDoS, and Fog Computing realm, the downside is the aptness of performance metric applicable to the proposed solutions. In evaluating the security solutions, the most important parameters that must be measured are comprehensiveness, accuracy, and performance. It is challenging and expensive to present the testimony for the exactitude [165] of cyber solutions. *Not all papers validate all the necessary parameters in the performance matrix.* The performance of the solution (some literature refers to it as defense strength) is based on the correct and incorrect classification of attack traffic as shown in Fig. 13.

The performance metric for DDoS/IoT-DDoS attacks should include (at least) various important performance measure parameters such as True Positives, True Negatives, False Positives, False Negatives, Recall, Precision, FI-score, Reliability, False Negative Rate, Accuracy, and Specificity [103,166].

**Table 7**
Traceability of benchmarks from existing defense solutions.

| Pack of 'What went well' | References |
|---|---|
| Delegation of IoT-DDoS traffic analysis out to cloud layer to underlying layers. | [96,129,142, 143,151, 152] |
| Use of Machine Learning and Deep Learning Algorithms for anomaly detection | [96,145,147, 150,155] |
| Use of Software-defined Network (SDN) to detect and mitigation DDoS/IoT-based DDoS Attacks. | [96,129, 141–148, 150] |
| Blockchain based distributed security and privacy | [148,154] |
| Defense solution based on traffic behavior patterns (Most suitable for IoT devices) | [157] |

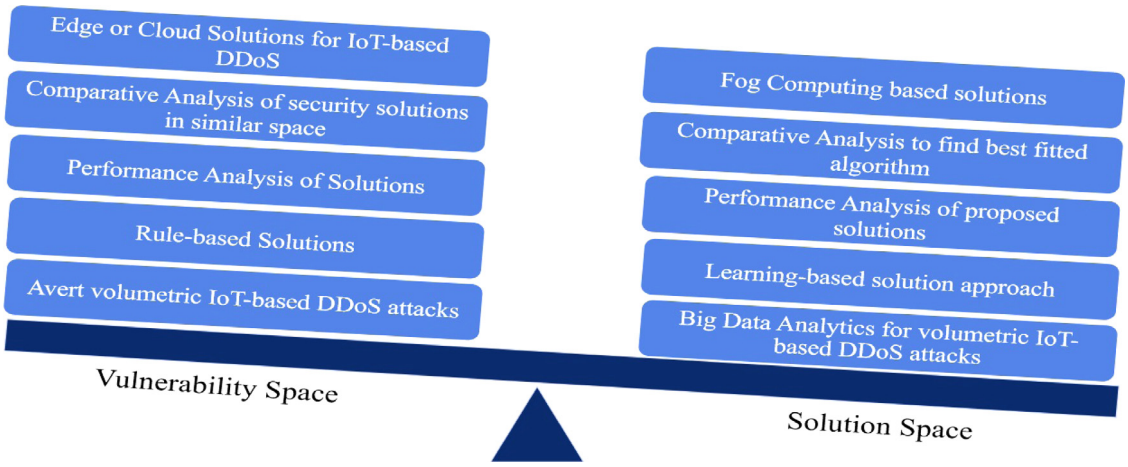


**Fig. 14.** Vulnerability Analysis Classification.

*7.2.2. Vulnerability analysis*

The vulnerability analysis from the literature survey is classified and presented in Fig. 14. The vulnerability space in the figure refers to the research gaps in existing solutions in the literature survey and solution space refers to technological bridges to overcome the research gaps.

Table 8 elucidates the details of vulnerability analysis carried out on the literature survey:

*7.2.3. Open research issues and challenges*

The Software-defined Cyber–Physical System has influenced the way for present-day smart systems. However, it has also brought several security challenges that needs immediate attention to the research community. The issues are listed below:

(a) *Need for handling IoT-generated colossal network traffic:* With the dawn and exponential increase in the use of IoT devices, the conventional DDoS mitigation techniques are not sufficient enough to offer an efficient security solution. The heterogeneous network of devices is diverse, large scale, and geographically distributed. Moreover, the present-day solutions do not take the high-volume real-time traffic stream into consideration. Unless accompanied by any of big-data like technologies, the solution execution performance is under question.

(b) *IoT-based DDoS Mitigation Techniques:* The issue of generating DDoS attacks using large scale IoT devices against Cyber–Physical Systems (CPS) is quite new and only an initial work exists for mitigating DDoS attacks initiated from the third-party devices.

(c) *Evaluation of solutions on simulated experimental testbeds:* Although plenty of solutions have been proposed by the research community, the exponential rise in attack size and frequency manifests the ineffectualness of the solutions. In most of the cases, researchers are replying upon simulated environment [96,129,157]. Hence, there is a dire need to invest a substantial amount of effort to device a real-time experimentation testbed for effective evaluation of the mitigation solutions.

(d) *Benchmark dataset availability for IoT-DDoS:* Most of the presented solution relies on either simulated traffic stream or refer to the datasets old as the hills. Standard dataset availability (labeled or unlabeled) is another task that needs to be taken care of.

(e) *Distributing Computing Power to Fog Nodes for IoT-based traffic:* In the light of previous research work, most of the proposed mitigation solutions are either provided at the edge of the network or at the cloud level in a Cyber–Physical System [144,157]. However, in case of legitimate traffic from dispersed IoT devices, it is difficult to detect the DDoS attacks at the edge computing level because of the sheer volume of the data traffic appearing to be or is originating from legitimate IoT devices. Providing a solution at the cloud computing level is expensive in terms of computational cost. Only a few researchers have concentrated on the distribution of the computational cost to different fog computing nodes.

(f) *DDoS Mitigation against SDN infrastructure:* The great majority of researchers have emphasized on providing DDoS detection and mitigation frameworks and/or solutions using SDN. However, only a few researchers have concentrated on SDN's vulnerabilities against DDoS attacks in Software-Defined Cyber–Physical Systems.

**Table 8**
Traceability of vulnerability class to Defense/Mitigation Solutions under study.

| Solution(s) referenced in study | Vulnerability class | Vulnerability in the proposed solutions |
|---|---|---|
| [96,145,147,150] | **Training and Execution Time** | The algorithm training and execution time is a crucial parameter for real-time intelligent machines. The aforesaid parameter remained unexplored in the proposed defense solutions. |
| [129,144–148,150,157] | **Inopportune deployment of defense solutions** | The proposed defense solutions are suggested either at edge computing or at the cloud computing layer. Edge computing is inappropriate for computationally complex and intelligent algorithms because of low processing nodes. The solutions deployed at the cloud layer may result in computational overhead to cloud services. The fog computing layer can be leveraged to handle volumetric traffic and latency reduction. |
| [10,96,129,143,144,146,150,151, 153,154,157] | **Validation of solutions against decrepit datasets** | The proposed solutions have been validated against old or simulated datasets. Therefore, there is a strong need to validate the solutions against contemporary high-volume, heterogeneous real-time network traffic. |
| [96,141,145–148,154] | **Sketchy discussion of Performance Metric(s)** | Due consideration has not been given to all the critical performance metrics such as True Positives (TP), True Negatives (TN), False Positives (FP), False Negatives (FN), Recall, Precision, FI-score, Reliability, False Negative Rate, Accuracy, and Specificity. |
| [96,150] | **Mitigating Zero-day attacks** | The proposed intelligent systems are based on supervised learning algorithms that are trained against labeled datasets. However, there is a strong need to shift the trend to unsupervised learning techniques for startling anomalies in modern-day network traffic to defense against zero-day attacks. |
| [141,142,157] | **Hypothetical models** | The papers propose only a Hypothetical model and does not discuss much about implementation and system performance. |
| [142,153] | **Real-time processing of enormous IoT traffic** | The proposed solutions lack the real-time processing of network traffic. The network traffic is increasing exponentially and is heterogeneous in nature. The contemporary big-data analytics tools need to be integrated with the existing defense solutions for real-time network traffic processing. |

## 8. Future scope and conclusion

In this paper, we have studied and examined the state-of-the-art Cyber–Physical System, components of modern-day Cyber–Physical System, architectural details, security issues with a focus on most devastating DDoS and IoT-DDoS attacks. Fog Computing has been proposed as a layer between perception and cloud for performance improvement and executing the delegated tasks on behalf of the cloud. We have studied the proposed solutions in the field of DDoS/IoT-DDoS detection and mitigation. Finally, we have carried out vulnerability and gap analysis of the available solutions and concluded to general gaps or vulnerabilities using the narrow down approach. We have, by means of this survey, have attempted to summarize the vulnerability analysis which can serve as a base for future technologists and researchers who are looking for developing defense solutions against DDoS/IoT-DDoS attacks. The vulnerability study is not limited in scope to DDoS attacks but has the applicability to more cybersecurity concerns as well.

Given the broad spectrum of scope for Cyber–Physical Systems, we further intend to present a solution against IoT-DDoS attacks leveraging Fog Computing capabilities in the Software-defined Cyber–Physical System because the discussed security concerns should be given the utmost attention and should be addressed aggressively. We will continue to behold remarkable bustle around Cyber–Physical Systems. The spring of supporting technologies and supplication areas in Cyber–Physical System requiring further research are, for example, Security as-a-service, application areas of CPS (Smart Cities, Smart Farming, etc.), Security Issues and Challenges in CPS (Smart Cities, Medical Cyber–Physical Systems, etc.), Intent-based networking in Software-defined Networks.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

The authors would like to pass-on gratitude to anonymous reviewers for their constructive feedback and valuable suggestions.

## References

[1] L. Hou, S. Zhao, X. Xiong, K. Zheng, P. Chatzimisios, M.S. Hossain, W. Xiang, Internet of things cloud: Architecture and implementation, IEEE Commun. Mag. 54 (12) (2016) 32–39, http://dx.doi.org/10.1109/MCOM.2016.1600398CM.
[2] S.H. Ahmed, G. Kim, D. Kim, Cyber physical system: Architecture, applications and research challenges, IFIP Wirel. Days (November 2013) (2013) http://dx.doi.org/10.1109/WD.2013.6686528.
[3] Y. Lu, Industry 4.0: A survey on technologies, applications and open research issues, J. Ind. Inf. Integr. 6 (2017) 1–10, http://dx.doi.org/10.1016/j.jii.2017.04.005.
[4] X. Wenfeng, W. Yonggang, H.F. Chuan, N. Dusit, X. Haiyong, A survey on software-defined networking, Asian Pacific J. Reprod. 7 (2) (2018) 72, http://dx.doi.org/10.4103/2305-0500.228016.
[5] B. Raghavan, T. Koponen, A. Ghodsi, Software-defined internet architecture: Decoupling architecture from infrastructure, in: Proceedings of the 11th ACM Workshop on Hot Topics in Networks, 2012, pp. 43–48.

[6] S.K. Khaitan, J.D. McCalley, Design techniques and applications of cyber physical systems: A survey, IEEE Syst. J. 9 (2) (2015) 350–365, http://dx.doi.org/10.1109/JSYST.2014.2322503.
[7] H. Kim, N. Feamster, Improving network management with software defined networking, IEEE Commun. Mag. 51 (2) (2013) 114–119, http://dx.doi.org/10.1109/MCOM.2013.6461195.
[8] W. Venters, E.A. Whitley, A critical review of cloud computing: Researching desires and realities, J. Inf. Technol. 27 (3) (2012) 179–197, http://dx.doi.org/10.1057/jit.2012.17.
[9] Elastic Load Balancing - Amazon Web Services, Amazon Web Services, Inc, 2020, https://aws.amazon.com/elasticloadbalancing/, (Accessed on 03 April 2020).
[10] B. Paharia, K. Bhushan, A comprehensive review of distributed denial of service (DDoS) attacks in fog computing environment, in: B.B. Gupta, G.M. Perez, D.P. Agrawal, D. Gupta (Eds.), Handbook of Computer Networks and Cyber Security: Principles and Paradigms, Springer International Publishing, Cham, 2020, pp. 493–524, http://dx.doi.org/10.1007/978-3-030-22277-2_20.
[11] Cyber-physical system - Wikipedia, 2020, https://en.wikipedia.org/wiki/Cyber-physical_system, (Accessed on 18 March 2020).
[12] H. Chen, Applications of cyber-physical system: A literature review, J. Ind. Integr. Manage. 02 (03) (2017) 1750012, http://dx.doi.org/10.1142/s2424862217500129.
[13] M. Yannuzzi, R. Milito, R. Serral-Gracia, D. Montero, M. Nemirovsky, Key ingredients in an IoT recipe: Fog computing, cloud computing, and more fog computing, in: 2014 IEEE 19th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks, CAMAD 2014, 2014, pp. 325–329, http://dx.doi.org/10.1109/CAMAD.2014.7033259.
[14] S. Sarkar, S. Misra, Theoretical modelling of fog computing: A green computing paradigm to support IoT applications, IET Netw. 5 (2) (2016) 23–29, http://dx.doi.org/10.1049/iet-net.2015.0034.
[15] S. Mansfield-Devine, DDoS: Threats and mitigation, Netw. Secur. (12) (2011) 5–12, http://dx.doi.org/10.1016/S1353-4858(11)70128-3.
[16] Constantinos. Kolias, Georgios. Kambourakis, Angelos. Stavrou, Jeffrey. Voas, DDoS in the IoT: Mirai and other botnets, Computer (2017) 1.
[17] G. Kambourakis, C. Kolias, A. Stavrou, The mirai botnet and the IoT zombie armies, in: Proceedings - IEEE Military Communications Conference MILCOM, Vol. 2017-Octob, 2017, pp. 267–272, http://dx.doi.org/10.1109/MILCOM.2017.8170867.
[18] Understanding the mirai botnet, USENIX Secur. (2017) 1093–1110, http://dx.doi.org/10.1016/j.religion.2008.12.001.
[19] N. Farah, M. Avishek, F. Muhammad, A. Rahman, M. Rafni, D. Md., Application of machine learning approaches in intrusion detection system: A survey, Int. J. Adv. Res. Artif. Intell. 4 (3) (2015) 9–18, http://dx.doi.org/10.14569/ijarai.2015.040302.
[20] C.F. Tsai, Y.F. Hsu, C.Y. Lin, W.Y. Lin, Intrusion detection by machine learning: A review, Expert Syst. Appl. 36 (10) (2009) 11994–12000, http://dx.doi.org/10.1016/j.eswa.2009.05.029.
[21] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, M.S. Lew, Intrusion detection by machine learning: A review, Neurocomputing 187 (10) (2016) 27–48, http://dx.doi.org/10.1016/j.neucom.2015.09.116.
[22] I.A.T. Hashem, I. Yaqoob, N.B. Anuar, S. Mokhtar, A. Gani, S. Ullah Khan, The rise of "big data" on cloud computing: Review and open research issues, Inf. Syst. 47 (2015) 98–115, http://dx.doi.org/10.1016/j.is.2014.07.006.
[23] C. Kacfah Emani, N. Cullot, C. Nicolle, Understandable big data: A survey, Comp. Sci. Rev. 17 (2015) 70–81, http://dx.doi.org/10.1016/j.cosrev.2015.05.002.
[24] P. O'Donovan, C. Gallagher, K. Bruton, D.T. O'Sullivan, A fog computing industrial cyber-physical system for embedded low-latency machine learning Industry 4.0 applications, Manuf. Lett. 15 (2018) 139–142, http://dx.doi.org/10.1016/j.mfglet.2018.01.005.
[25] A. Čolaković, M. Hadžialić, Internet of Things (IoT): A review of enabling technologies, challenges, and open research issues, Comput. Netw. 144 (2018) 17–39, http://dx.doi.org/10.1016/j.comnet.2018.07.017.
[26] R. Kumar, R. Goyal, On cloud security requirements, threats, vulnerabilities and countermeasures: A survey, Comp. Sci. Rev. 33 (2019) 1–48, http://dx.doi.org/10.1016/j.cosrev.2019.05.002.
[27] Empowering App Development for Developers, Docker, 2020, https://www.docker.com/, (Accessed on 12 June 2020).
[28] Infrastructure for container projects, 2020, https://linuxcontainers.org, (Accessed on 12 June 2020).
[29] CRIU, 2020, https://criu.org/Main_Page, (Accessed on 12 June 2020).
[30] Openflow. Open networking foundation (ONF), 2019, https://www.opennetworking.org/, (Accessed on 15 July 2019).
[31] B. Fraser, D. Lake, J. Finnegan, N. Viljoen, M. Miller, N. Rao, Are we ready for SDN? Implementation challenges for software-defined networks, IEEE Commun. Mag. (July) (2013) 36–43.
[32] D. Kreutz, F.M. Ramos, P.E. Verissimo, C.E. Rothenberg, S. Azodolmolky, S. Uhlig, Software-defined networking: A comprehensive survey, Proc. IEEE 103 (1) (2015) 14–76, http://dx.doi.org/10.1109/JPROC.2014.2371999.
[33] Forwarding and Control Element Separation (ForCES) protocol specification, 2019, https://www.ietf.org/rfc/rfc5810.txt, (Accessed on 12 July 2019).
[34] ONOS - an open source controller, 2020, https://onosproject.org/, (Accessed on 31 March 2020).
[35] ONOS-Wiki, 2020, https://wiki.onosproject.org/display/ONOS/Wiki+Home, (Accessed on 31 March 2020).
[36] Ryu - an open source controller, 2020, https://osrg.github.io/ryu/, (Accessed on 31 March 2020).
[37] Home - opendaylight, 2020, https://www.opendaylight.org/, (Accessed on 31 March 2020).
[38] Opendaylight-wiki, 2020, https://wiki.opendaylight.org/view/Main_Page, (Accessed on 31 March 2020).
[39] POX - an open source controller, 2020, https://github.com/noxrepo/pox, (Accessed on 31 March 2020).
[40] HPE support center | HPE van SDN controller software, 2020, https://support.hpe.com/hpesc/public/docDisplay?docId=emr_na-c03967699#N10012, (Accessed on 31 March 2020).
[41] Huawei software defined network (SDN) solution, 2020, https://actfornet.com/huawei-cloud/sdn.html, (Accessed on 31 March 2020).
[42] Open networking foundation - wikipedia, in: Wikipedia, 2020, https://en.wikipedia.org/wiki/Open_Networking_Foundation, (Accessed on 20 February 2020).
[43] C. Yoon, T. Park, S. Lee, H. Kang, S. Shin, Z. Zhang, Enabling security functions with SDN: A feasibility study, Comput. Netw. 85 (2015) 19–35, http://dx.doi.org/10.1016/j.comnet.2015.05.005.
[44] S. Jain, M. Zhu, J. Zolla, U. Hölzle, S. Stuart, A. Vahdat, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu, J. Zolla, U. Hölzle, S. Stuart, A. Vahdat, B4 - experience with a globally deployed SDWAN, in: Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM, Vol. 43, SIGCOMM '13, (4) 2013, p. 3, http://dx.doi.org/10.1145/2486001.2486019.
[45] CISCO, Software-defined networking: Why we like it and how we are building on it, in: White Paper, 2013, pp. 1–4, https://www.cisco.com/c/dam/en_us/solutions/industries/docs/gov/cis13090_sdn_sled_white_paper.pdf, (Accessed on 02 august 2020).
[46] The VMware NSX network virtualization platform, in: TECHNICAL WHITE PAPER, 2013, (Accessed on 11 june 2020), https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/whitepaper/products/nsx/vmware-nsx-network-virtualization-platform-white-paper.pdf.
[47] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, J. Turner, Openflow: Enabling innovation in campus networks, SIGCOMM Comput. Commun. Rev. 38 (2) (2008) 69–74, http://dx.doi.org/10.1145/1355734.1355746.
[48] R. Enns, RFC 4741 - NETCONF configuration protocol, in: Internet Draft, Internet Engineering Task Force, 2006, https://tools.ietf.org/html/rfc4741, (Accessed on 31 March 2020).
[49] D. Hasan, M. Othman, Efficient topology discovery in software defined networks: Revisited, Procedia Comput. Sci. 116 (2017) 539–547, http://dx.doi.org/10.1016/j.procs.2017.10.051.
[50] Openflow management and configuration protocol (OF-CONFIG) v1.2, in: Open Networking Foundation, ONF TS-016, 2014, https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow-config/of-config-1.2.pdf, (Accessed on 11 june 2020).
[51] B. Pfaff, B. Davie, The open vswitch database management protocol, in: RFC 7047 (Informational), Internet Engineering Task Force, 2013, https://www.ietf.org/rfc/rfc7047.txt, (Accessed on 22 january 2020).
[52] What is REST – learn to create timeless REST APIs, 2020, https://restfulapi.net/, (Accessed on 31 january 2020).
[53] M.P. Singh, A. Bhandari, New-flow based DDoS attacks in SDN: Taxonomy, rationales, and research challenges, Comput. Commun. 154 (2020) 509–527, http://dx.doi.org/10.1016/j.comcom.2020.02.085.
[54] Software defined networking (SDN) market research report- global forecast 2023, 2019, https://www.marketresearchfuture.com/reports/software-defined-networking-market-1607, (Accessed on 01 august 2019).
[55] Openflow-enabled SDN and network functions virtualization, in: Open Networking Foundation, 2014, https://www.opennetworking.org/images/stories/downloads/sdn-resources/solution-briefs/sb-sdn-nvf-solution.pdf, (Accessed on 18 March 2020).
[56] M.S. Bonfim, K.L. Dias, S.F.L. Fernandes, C. De, Integrated NFV / SDN Architectures : A Systematic Review, Vol. 2, (3) 2010.
[57] R. Szabó, M. Kind, F.J. Westphal, H. Woesner, D. Jocha, A. Császar, Elastic network functions: Opportunities and challenges, IEEE Netw. 29 (3) (2015) 15–21, http://dx.doi.org/10.1109/MNET.2015.7113220.

[58] R. Mijumbi, J. Serrat, J.L. Gorricho, N. Bouten, F. De Turck, R. Boutaba, Network function virtualization: State-of-the-art and research challenges, IEEE Commun. Surv. Tutor. 18 (1) (2016) 236–262, http://dx.doi.org/10.1109/COMST.2015.2477041, arXiv:1509.07675.

[59] Network function virtualization (NFV): Architectural framework v1.1.1, in: ETSI, Tech. Rep, ETSI GS NFV 002, 2013, https://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/01.01.01_60/gs_NFV002v010101p.pdf, (Accessed on 20 February 2020).

[60] ETSI - standards for NFV - network functions virtualisation | nfv solutions, 2020, https://www.etsi.org/technologies/nfv, (Accessed on 20 February 2020).

[61] Wiki - internet of things, 2019, https://en.wikipedia.org/wiki/Internet{_}of{_}things, (Accessed on 25 june 2019).

[62] Wiki - cloud computing, 2019, https://en.wikipedia.org/wiki/Cloud{_}computing, (Accessed on 25 june 2019).

[63] F. Tao, M. Zhang, A. Nee, Digital twin, cyber–physical system, and internet of things, in: Digital Twin Driven Smart Manufacturing, 2019, pp. 243–256, http://dx.doi.org/10.1016/b978-0-12-817630-6.00012-6.

[64] C. Alippi, S. Ozawa, Computational intelligence in the time of cyber-physical systems and the internet of things, in: Artificial Intelligence in the Age of Neural Networks and Brain Computing, Elsevier Inc., 2019, pp. 245–263, http://dx.doi.org/10.1016/b978-0-12-815480-9.00012-8.

[65] S. Majumder, A. Mathur, A.Y. Javaid, Retraction Note to: Cyber-Physical System Security Controls: A Review, 2019, http://dx.doi.org/10.1007/978-3-319-92564-6_9.

[66] T. Sanislav, L. Miclea, Cyber-physical systems - concept, challenges and research areas, Control Eng. Appl. Inf. 14 (2) (2012) 28–33.

[67] C. Greer, M. Burns, D. Wollman, E. Griffor, Cyber-physical systems and internet of things NIST special publication 1900-202 cyber-physical systems and internet of things, 2019.

[68] M. Faheem, S.B. Shah, R.A. Butt, B. Raza, M. Anwar, M.W. Ashraf, M.A. Ngadi, V.C. Gungor, Smart grid communication and information technologies in the perspective of industry 4.0: Opportunities and challenges, Comp. Sci. Rev. 30 (2018) 1–30, http://dx.doi.org/10.1016/j.cosrev.2018.08.001.

[69] A. Smiti, When machine learning meets medical world : Current status and future challenges, Comp. Sci. Rev. 37 (2020) 100280, http://dx.doi.org/10.1016/j.cosrev.2020.100280.

[70] J.I. Jimenez, H. Jahankhani, S. Kendzierskyj, Health Care in the Cyberspace: Medical Cyber-Physical System and Digital Twin Challenges, Springer International Publishing, 2020, pp. 79–92, http://dx.doi.org/10.1007/978-3-030-18732-3_6.

[71] S. ULLAH, P. KHAN, N. ULLAH, S. SALEEM, H. HIGGINS, K. Sup KWAK, A review of wireless body area networks for medical applications, Int. J. Commun. Netw. Syst. Sci. 02 (08) (2009) 797–803, http://dx.doi.org/10.4236/ijcns.2009.28093, arXiv:1001.0831.

[72] A.M. Rahmani, T.N. Gia, B. Negash, A. Anzanpour, I. Azimi, M. Jiang, P. Liljeberg, Exploiting smart e-health gateways at the edge of healthcare Internet-of-Things: A fog computing approach, Future Gener. Comput. Syst. 78 (2018) 641–658, http://dx.doi.org/10.1016/j.future.2017.02.014.

[73] J. Winkley, P. Jiang, W. Jiang, Verity: An ambient assisted living platform, IEEE Trans. Consum. Electron. 58 (2) (2012) 364–373, http://dx.doi.org/10.1109/TCE.2012.6227435.

[74] T.H. Uhlemann, C. Lehmann, R. Steinhilper, The digital twin: Realizing the cyber-physical production system for industry 4.0, Proc. CIRP 61 (2017) 335–340, http://dx.doi.org/10.1016/j.procir.2016.11.152.

[75] A Cyber-Physical Systems architecture for industry 4.0-based manufacturing systems, Manuf. Lett. 3 (2015) 18–23, http://dx.doi.org/10.1016/j.mfglet.2014.12.001.

[76] Y.-d. Lin, N. Chiao, Standardization for cloud, (November) 2014, pp. 19–21.

[77] E. Molina, E. Jacob, Software-defined networking in cyber-physical systems: A survey, Comput. Electr. Eng. 66 (2018) 407–419, http://dx.doi.org/10.1016/j.compeleceng.2017.05.013.

[78] IoT: number of connected devices worldwide 2012-2025 | statista, in: Statista Research Department, 2020, https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/, (Accessed on 22 march 2020).

[79] Cloud computing trends: 2019 state of the cloud survey | Flexera blog, in: Flexera, 2019, https://www.flexera.com/blog/cloud/2019/02/cloud-computing-trends-2019-state-of-the-cloud-survey/, (Accessed on 17 march 2020).

[80] T. Coughlin, Forbes - 175 Zettabytes By 2025, Forbes Media LLC, 2018, https://www.forbes.com/sites/tomcoughlin/2018/11/27/175-zettabytes-by-2025/{#}6c8160635459, (Accessed on 20 march 2020).

[81] Q. Yan, F.R. Yu, Q. Gong, J. Li, IEEE Commun. Surv. Tutor. (2015) http://dx.doi.org/10.1109/COMST.2015.2487361.

[82] S. Sibi Chakkaravarthy, D. Sangeetha, V. Vaidehi, A survey on malware analysis and mitigation techniques, Comp. Sci. Rev. 32 (2019) 1–23, http://dx.doi.org/10.1016/j.cosrev.2019.01.002, https://doi.org/10.1016/j.cosrev.2019.01.002.

[83] Q. Covert, D. Steinhagen, M. Francis, K. Streff, Towards a triad for data privacy, in: Proceedings of the 53rd Hawaii International Conference on System Sciences, Vol. 3, 2020, pp. 4379–4387, http://dx.doi.org/10.24251/hicss.2020.535.

[84] A. Bakr, A.A. Abd El-Aziz, H.A. Hefny, A survey on mitigation techniques against ddos attacks on cloud computing architecture, Int. J. Adv. Sci. Technol. 28 (12) (2019) 187–200.

[85] Information security - wikipedia, in: Wikipedia, 2020, https://en.wikipedia.org/wiki/Information_security#Key_concepts, (Accessed on 20 February 2020).

[86] I. Farris, T. Taleb, Y. Khettab, J. Song, A survey on emerging SDN and NFV security mechanisms for IoT systems, IEEE Commun. Surv. Tutor. 21 (1) (2019) 812–837, http://dx.doi.org/10.1109/COMST.2018.2862350.

[87] A. Mosenia, N.K. Jha, A comprehensive study of security of internet-of-things, IEEE Trans. Emerg. Top. Comput. 5 (4) (2017) 586–602, http://dx.doi.org/10.1109/TETC.2016.2606384.

[88] B. Parno, A. Perrig, V. Gligor, Distributed detection of node replication attacks in sensor networks, 2005, pp. 49–63, http://dx.doi.org/10.1109/sp.2005.8.

[89] G. Somani, M.S. Gaur, D. Sanghi, M. Conti, R. Buyya, DDoS attacks in cloud computing: Issues, taxonomy, and future directions, Comput. Commun. 107 (2017) 30–48, http://dx.doi.org/10.1016/j.comcom.2017.03.010, arXiv:1512.08187.

[90] V.T. Dang, T.T. Huong, N.H. Thanh, P.N. Nam, N.N. Thanh, A. Marshall, SDN-based SYN proxy - A solution to enhance performance of attack mitigation under TCP SYN Flood, Comput. J. 62 (4) (2019) 518–534, http://dx.doi.org/10.1093/comjnl/bxy117.

[91] S. Scott-Hayward, G. O'Callaghan, S. Sezer, SDN security: A survey, in: SDN4FNS 2013 - 2013 Workshop on Software Defined Networks for Future Networks and Services, 2013, http://dx.doi.org/10.1109/SDN4FNS.2013.6702553.

[92] R. Swami, M. Dave, V. Ranga, Software-defined networking-based DDoS defense mechanisms, ACM Comput. Surv. 52 (2) (2019) http://dx.doi.org/10.1145/3301614.

[93] E. Marín-Tordera, X. Masip-Bruin, J. García-Almiñana, A. Jukan, G.J. Ren, J. Zhu, Do we all really know what a fog node is? Current trends towards an open definition, Comput. Commun. 109 (2017) 117–130, http://dx.doi.org/10.1016/j.comcom.2017.05.013.

[94] M. Chiang, T. Zhang, Fog and IoT: An overview of research opportunities, IEEE Internet Things J. 3 (6) (2016) 854–864, http://dx.doi.org/10.1109/JIOT.2016.2584538.

[95] K.H. Abdulkareem, M.A. Mohammed, S.S. Gunasekaran, M.N. Al-Mhiqani, A.A. Mutlag, S.A. Mostafa, N.S. Ali, D.A. Ibrahim, A review of fog computing and machine learning: Concepts, applications, challenges, and open issues, IEEE Access 7 (2019) 153123–153140, http://dx.doi.org/10.1109/ACCESS.2019.2947542.

[96] R. Priyadarshini, R.K. Barik, A deep learning based intelligent framework to mitigate DDoS attack in fog environment, J. King Saud Univ. - Comput. Inf. Sci. (2019) http://dx.doi.org/10.1016/j.jksuci.2019.04.010.

[97] S. Khan, S. Parkinson, Y. Qin, Fog computing security: a review of current applications and security solutions, J. Cloud Comput. 6 (1) (2017) 19, http://dx.doi.org/10.1186/s13677-017-0090-3.

[98] W. Shi, J. Cao, Q. Zhang, Y. Li, L. Xu, Edge computing: Vision and challenges, IEEE Internet Things J. 3 (5) (2016) 637–646, http://dx.doi.org/10.1109/JIOT.2016.2579198.

[99] S. Behal, K. Kumar, M. Sachdeva, D-FACE: An anomaly based distributed approach for early detection of DDoS attacks and flash events, J. Netw. Comput. Appl. 111 (2018) 49–63, http://dx.doi.org/10.1016/j.jnca.2018.03.024.

[100] Fog computing and the internet of things: Extend the cloud to where the things are, 2018, https://www.cisco.com/c/dam/en{_}us/solutions/trends/iot/docs/computing-overview.pdf, (Accessed on 01 august 2019).

[101] M. Bernard, That's data science: Airbus puts 10,000 sensors in every single wing!, 2015, https://www.datasciencecentral.com/profiles/blogs/that-s-data-science-airbus-puts-10-000-sensors-in-every-single, (Accessed on 01 august 2019).

[102] B. Butler, Cisco brings its SDN to Amazon, Microsoft and Google's public cloud, 2019, https://www.networkworld.com/article/3218045/cisco-brings-its-sdn-to-amazon-microsoft-and-google-s-public-cloud.html, (Accessed on 03 august 2019).

[103] Resource library imperva | the anatomy of a DDoS attack, 2020, https://www.imperva.com/resources/resource-library/infographics/the-anatomy-of-a-ddos-attack/, (Accessed on 03 february 2020).

[104] Distributed-denial-of-service attacks - wiki, 2019, https://en.wikipedia.org/wiki/Denial-of-service_attack, (Accessed on 02 july 2019).

[105] S.T. Zargar, J. Joshi, D. Tipper, A survey of defense mechanisms against distributed denial of service (DDOS) flooding attacks, IEEE Commun. Surv. Tutor. 15 (4) (2013) 2046–2069, http://dx.doi.org/10.1109/SURV.2013.031413.00127.

[106] K.S. Sahoo, S.K. Panda, S. Sahoo, B. Sahoo, R. Dash, Toward secure software-defined networks against distributed denial of service attack, J. Supercomput. 75 (8) (2019) 4829–4874, http://dx.doi.org/10.1007/s11227-019-02767-z.
[107] S. Kumar, M. Azad, O. Gomez, R. Valdez, Can microsoft's Service Pack2 (SP2) security software prevent SMURF attacks? in: Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services, Vol. 2006, AICT/ICIW'06, 2006, p. 89, http://dx.doi.org/10.1109/AICT-ICIW.2006.60.
[108] Z. Trabelsi, S. Zeidan, K. Hayawi, Denial of firewalling attacks (DoF): The case study of the emerging blacknurse attack, IEEE Access 7 (2019) 61596–61609, http://dx.doi.org/10.1109/ACCESS.2019.2915792.
[109] O. Sheeba, N. Vinayan, A survey on characterization of defense mechanisms in DDOS attacks, Int. J. Eng. Adv. Technol. 97 (2013) 321–324.
[110] A. Bhandari, A.L. Sangal, K. Kumar, Characterizing flash events and ddos attacks - an empirical investigation, in: Int. J. Appl. Eng. Res., 9, (22) 2014, pp. 5968–5974, http://dx.doi.org/10.1002/sec.1472, arXiv:0806.0557.
[111] A. Bhandari, A. Sangal, K. Kumar, Destination address entropy based detection and traceback approach against distributed denial of service attacks, Int. J. Comput. Netw. Inf. Secur. 7 (8) (2015) 9–20, http://dx.doi.org/10.5815/ijcnis.2015.08.02.
[112] I. Sreeram, V.P.K. Vuppala, HTTP flood attack detection in application layer using machine learning metrics and bio inspired bat algorithm, Appl. Comput. Inf. 15 (1) (2019) 59–66, http://dx.doi.org/10.1016/j.aci.2017.10.003.
[113] S. Behal, K. Kumar, M. Sachdeva, Characterizing ddos attacks and flash events: Review, research gaps and future directions, Comp. Sci. Rev. 25 (2017) 101–114, http://dx.doi.org/10.1016/j.cosrev.2017.07.003.
[114] T. Peng, C. Leckie, K. Ramamohanarao, Survey of network-based defense mechanisms countering the DoS and DDoS problems, ACM Comput. Surv. 39 (1) (2007) http://dx.doi.org/10.1145/1216370.1216373.
[115] A. Febro, H. Xiao, J. Spring, Distributed SIP ddos defense with P4, in: IEEE Wireless Communications and Networking Conference, Vol. 2019-April, WCNC, IEEE, 2019, pp. 1–8, http://dx.doi.org/10.1109/WCNC.2019.8885926.
[116] S. Majumder, A. Mathur, A.Y. Javaid, Retraction Note to: Cyber-Physical System Security Controls: A Review, 2019, p. C1, http://dx.doi.org/10.1007/978-3-319-92564-6_9.
[117] H. Kaur, S. Behal, K. Kumar, Characterization and comparison of distributed denial of service attack tools, in: Proceedings of the 2015 International Conference on Green Computing and Internet of Things, ICGCIoT 2015, 2016, pp. 1139–1145, http://dx.doi.org/10.1109/ICGCIoT.2015.7380634.
[118] Q3 2019 cyber threats and trends report | neustar, 2020, https://www.home.neustar/resources/whitepapers/2019-cyber-threats-trends-report-q3, (Accessed on 21 march 2020).
[119] A. Worldwide, I.S. Report, cloud in the crosshairs, 2019, https://www.netscout.com/sites/default/files/2019-03/SECR_005_EN-1901--WISR.pdf, (Accessed on 01 august 2019).
[120] DDoS & Network Visibility Solutions, NETSCOUT Arbor, 2020, https://www.netscout.com/arbor-ddos, (Accessed on 12 march 2020).
[121] Security, Cloud Delivery, Performance, Akamai, 2020, https://www.akamai.com/, (Accessed on 12 march 2020).
[122] Cyber Security Leader, Imperva, Inc., 2020, https://www.imperva.com/, (Accessed on 12 march 2020).
[123] Trusted Connections at the Moments that Matter the Most, Neustar, 2020, https://www.home.neustar/, (Accessed on 12 march 2020).
[124] Krebsonsecurity hit with record DDoS, 2016, https://krebsonsecurity.com/2016/09/krebsonsecurity-hit-with-record-ddos/, (Accessed on 03 august 2019).
[125] B. Herzberg, I. Zeifman, D. Bekerman, Breaking down mirai: An IoT DDoS botnet analysis, 2016, https://www.imperva.com/blog/malware-analysis-mirai-ddos-botnet/, (Accessed on 03 august 2019).
[126] Projectshield by google and jigsaw, 2020, https://projectshield.withgoogle.com/landing, (Accessed on 02 august 2020).
[127] DDoS protection service, anti DDoS mitigation, cloudflare, 2020, https://www.cloudflare.com/ddos/, (Accessed on 02 august 2020).
[128] DDoS Mitigation, Akamai, 2020, https://www.akamai.com/us/en/resources/ddos-mitigation.jsp, (Accessed on 02 august 2020).
[129] M. Ozcelik, N. Chalabianloo, G. Gur, Software-defined edge defense against IoT-based DDoS, in: IEEE CIT 2017 - 17th IEEE International Conference on Computer and Information Technology, 2017, pp. 308–313, http://dx.doi.org/10.1109/CIT.2017.61.
[130] The economic times internet report - august 2019, in: The Economic Times Internet Report, 2019, https://economictimes.indiatimes.com/tech/internet/cyberattacks-grew-22-on-indias-iot-deployments-in-q2/articleshow/70606639.cms, (Accessed on 10 august 2019).
[131] HP study reveals 70 percent of internet of things devices vulnerable to attack, 2019, https://www8.hp.com/us/en/hp-news/press-release.html?id=1744676, (Accessed on 05 august 2019).
[132] S. Stein, J. Jacobs, Cyberattack hits HHS during coronavirus response - bloomberg, in: Bloomberg, 2020, https://www.bloomberg.com/news/articles/2020-03-16/u-s-health-agency-suffers-cyber-attack-during-covid-19-response, (Accessed on 20 march 2020).
[133] C.B.R. E.D. Targett Editor, Record DDoS attack hits AWS: 2.3 tbps assault lasted days, 2020, https://www.cbronline.com/news/record-ddos-attack-aws, (Accessed on 13 june 2020).
[134] K. Oleg, E. Badovskaya, A. Gutnikov, DDoS attacks in Q1 2019, 2019, https://securelist.com/ddos-report-q1-2019/90792/, (Accessed on 10 august 2019).
[135] NETSCOUT threat intellegence report 1H 2019, 2019, https://www.netscout.com/sites/default/files/2019-07/SECR_010_EN-1901, (Accessed on 12 march 2020).
[136] K. Oleg, S. Jens, K. Alexander, Ddos attacks Q3 2016, 2016, https://securelist.com/kaspersky-ddos-intelligence-report-for-q3-2016/76464/, (Accessed on 28 july 2019).
[137] S. Bjarnason, R. Dobbins, Evolution of a New DDoS Technique | NETSCOUT, NETSCOUT Systems, Inc., 2020, https://www.netscout.com/blog/asert/evolution-new-ddos-technique, (Accessed on 06 april 2020).
[138] K. Alexander, K. Oleg, B. Ekaterina, DDoS attacks in Q1 2018, 2019, https://securelist.com/ddos-report-in-q1-2018/85373/, (Accessed on 04 august 2019).
[139] K. Alexander, Khalimonenko Oleg, I. Kirill, DDoS attacks in Q4 2017, 2019, https://securelist.com/ddos-attacks-in-q4-2017/83729/, (Accessed on 10 august 2019).
[140] J.P.A. Yaacoub, O. Salman, H.N. Noura, N. Kaaniche, A. Chehab, M. Malli, Cyber-physical systems security: Limitations, issues and future trends, Microprocess. Microsyst. 77 (2020) 103201, http://dx.doi.org/10.1016/j.micpro.2020.103201.
[141] T. Karthick, M. Manikandan, Fog assisted IoT based medical cyber system for cardiovascular diseases affected patients, Concurr. Comput. 31 (12) (2019) 1–9, http://dx.doi.org/10.1002/cpe.4861.
[142] Q. Yan, W. Huang, X. Luo, Q. Gong, F.R. Yu, A multi-level DDoS mitigation framework for the industrial internet of things, IEEE Commun. Mag. 56 (2) (2018) 30–36, http://dx.doi.org/10.1109/MCOM.2018.1700621.
[143] K. Bhardwaj, J.C. Miranda, A. Gavrilovska, Towards IoT-DDoS prevention using edge computing, in: USENIX Workshop on Hot Topics in Edge Computing, HotEdge 18, 2018.
[144] N.Z. Bawany, J.A. Shamsi, SEAL: SDN based secure and agile framework for protecting smart city applications from DDoS attacks, J. Netw. Comput. Appl. 145 (2019) 102381, http://dx.doi.org/10.1016/j.jnca.2019.06.001.
[145] Q. Niyaz, W. Sun, A.Y. Javaid, A deep learning based ddos detection system in software-defined networking (SDN), ICST Trans. Secur. Saf. 4 (12) (2017) 153515, http://dx.doi.org/10.4108/eai.28-12-2017.153515.
[146] C. Buragohain, N. Medhi, Flowtrapp: An SDN based architecture for ddos attack detection and mitigation in data centers, in: 3rd International Conference on Signal Processing and Integrated Networks, SPIN 2016, 2016, pp. 519–524, http://dx.doi.org/10.1109/SPIN.2016.7566750.
[147] T.M. Nam, P.H. Phong, T.D. Khoa, T.T. Huong, P.N. Nam, N.H. Thanh, L.X. Thang, P.A. Tuan, L.Q. Dung, V.D. Loi, Self-organizing map-based approaches in ddos flooding detection using SDN, in: International Conference on Information Networking, Vol. 2018-Janua, 2018, pp. 249–254, http://dx.doi.org/10.1109/ICOIN.2018.8343119.
[148] P.K. Sharma, M.Y. Chen, J.H. Park, A software defined fog node based distributed blockchain cloud architecture for IoT, IEEE Access 6 (c) (2018) 115–124, http://dx.doi.org/10.1109/ACCESS.2017.2757955.
[149] Q. Wang, M. Su, Integrating blockchain technology into the energy sector —from theory of blockchain to research and application of energy blockchain, Comp. Sci. Rev. 37 (2020) 100275, http://dx.doi.org/10.1016/j.cosrev.2020.100275.
[150] S.S. Bhunia, M. Gurusamy, Dynamic attack detection and mitigation in IoT using SDN, in: 2017 27th International Telecommunication Networks and Applications Conference, Vol. 2017-Janua, ITNAC 2017, 2017, pp. 1–6, http://dx.doi.org/10.1109/ATNAC.2017.8215418.
[151] Deepali, K. Bhushan, DDoS attack defense framework for cloud using fog computing, in: RTEICT 2017 - 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, Proceedings, Vol. 2018-Janua, 2018, pp. 534–538, http://dx.doi.org/10.1109/RTEICT.2017.8256654.
[152] B. Paharia, K. Bhushan, Fog computing as a defensive approach against distributed denial of service (DDoS): A proposed architecture, in: 2018 9th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2018, IEEE, 2018, pp. 1–7, http://dx.doi.org/10.1109/ICCCNT.2018.8494060.
[153] L. Zhou, H. Guo, G. Deng, A fog computing based approach to DDoS mitigation in IIoT systems, Comput. Secur. 85 (2019) 51–62, http://dx.doi.org/10.1016/j.cose.2019.04.017.
[154] A. Dorri, S.S. Kanhere, R. Jurdak, P. Gauravaram, Blockchain for IoT security and privacy: The case study of a smart home, in: 2ND IEEE PERCOM Workshop On Security Privacy And Trust In The Internet of Things 2017, 2017, http://dx.doi.org/10.1109/PERCOMW.2017.7917634.

[155] I. Ko, D. Chambers, E. Barrett, Feature dynamic deep learning approach for DDoS mitigation within the ISP domain, Int. J. Inf. Secur. 19 (1) (2020) 53–70, http://dx.doi.org/10.1007/s10207-019-00453-y.
[156] Github - markus-go/bonesi: BoNeSi - the DDoS botnet simulator, in: Markus Goldstein, 2020, https://github.com/Markus-Go/bonesi, (Accessed on 20 march 2020).
[157] I. Cvitić, D. Peraković, M. Periša, M. Botica, Novel approach for detection of IoT generated DDoS traffic, Wirel. Netw. 1 (2019) http://dx.doi.org/10.1007/s11276-019-02043-1.
[158] I. Cvitić, D. Peraković, M. Periša, M. Botica, Smart home IoT traffic characteristics as a basis for DDoS traffic detection, 2018, http://dx.doi.org/10.4108/eai.6-11-2018.2279336.
[159] Amazon Web Services (AWS) - Cloud Computing Services, Amazon Web Services, Inc, 2020, https://aws.amazon.com/, (Accessed on 02 april 2020).
[160] Denial of Service Attack Mitigation on AWS, Amazon Web Services, Inc., 2020, (Accessed on 03 April 2020), Denial of Service Attack Mitigation on AWS.
[161] AWS Best Practices for DDoS Resiliency, Amazon Web Services, Inc, 2019, https://d0.awsstatic.com/whitepapers/Security/DDoS_White_Paper.pdf, (Accessed on 03 April 2020).
[162] AWS WAF - web application firewall - amazon web services (AWS), 2020, https://aws.amazon.com/waf/, (Accessed on 02 april 2020).
[163] Amazon Route 53 - Amazon Web Services, Amazon Web Services, Inc, 2020, https://aws.amazon.com/route53/, (Accessed on 03 April 2020).
[164] AWS Shield - Amazon Web Services (AWS), Amazon Web Services, Inc, 2020, https://aws.amazon.com/shield/, (Accessed on 03 April 2020).
[165] A. Patcha, J.M. Park, An overview of anomaly detection techniques: Existing solutions and latest technological trends, Comput. Netw. 51 (12) (2007) 3448–3470, http://dx.doi.org/10.1016/j.comnet.2007.02.001.
[166] C. Xu, H. Lin, Y. Wu, X. Guo, W. Lin, An SDNFV-based ddos defense technology for smart cities, IEEE Access 7 (2019) 137856–137874, http://dx.doi.org/10.1109/ACCESS.2019.2943146.