Journal of Big Data

SURVEY PAPER Open Access

A survey of methods supporting cyber situational awareness in the context of smart cities

Nataliia Neshenko[1*], Christelle Nader[2], Elias Bou-Harb[2] and Borko Furht[1]

*Correspondence:
nneshenko2016@fau.edu
[1] Department of Computer and Electrical Engineering and Computer Sciences, Florida Atlantic University, Boca Raton, USA
Full list of author information is available at the end of the article

Abstract

A modern urban infrastructure no longer operates in isolation, but instead, leverages the latest technologies to collect, process, and distribute aggregated knowledge in order to improve the quality of the provided services and promote the efficiency of resource consumption. This technological development, however, manifests in the form of new vulnerabilities and a plethora of attack vectors. In the same context, the ambiguity of ever-evolving cyber threats and their debilitating consequences introduce new barriers for decision-makers. Therefore, cyber situational awareness of smart cities emerges as a mission-critical task that requires support methods for effective and timely decision-making. In this article, we investigate the threat landscape of smart cities, survey and reveal the progress in data-driven methods for situational awareness and evaluate their effectiveness when addressing various cyber threats. We draw several potential research directions that aim at advancing cyber situational awareness in the context of smart cities.

Keywords: Cyber analytics, Decision support models, Data science, Knowledge discovery, Machine learning, Applied analytics, Decision-making, Smart cities, Internet of things

Introduction

The United Nations predicts that two-thirds of the world population will live in urban areas [1] by 2050; implying that around 1.5 million people around the globe will move into a city every week [2]. This rapid growth comes with a myriad of challenges and opportunities. In fact, we witness extensive development in new infrastructure to support such large population to meet their environmental, social, and economic goals. From improving traffic conditions to optimizing energy consumption, smart cities enhance the quality of life of their residents by reducing carbon emission while optimizing utility costs.

With the embedding of latest technologies varying from telecommunication-enablers to advances in data-driven artificial intelligence, Internet-connected municipal infrastructure, parking meters, and alike, continue to collect and analyze data to support decision-making and pinpoint deficiencies for real-time optimization. For instance,

cities achieve a remarkable reduction of costs in heating, ventilation, and air conditioning (HVAC) by placing Internet-of-Things (IoT)-powered cooling systems to optimize HVAC usage based on the activities in each room [3]. Additionally, a smart grid is directly linked to resource efficient provisioning solutions that supports smart cities' sustainable environment goals. Moreover, evolving technology empowered by advances in shallow and deep learning analyzes patterns in city-wide energy consumption to only delivers an optimal amount. Further, cities use sensors to detect pipe leaks; New York city saved more than $73 million in water costs by notifying residents about possible (predicted) water leaks [4]. The latter becomes possible after the deployment of smart water meters and by exploiting advanced data analysis algorithms. Additionally, smart cities continue to maintain safety of their citizens. In fact, by using a system of connected video feeds, the city of Rio de Janeiro has improved the response time of emergency [5] and Chicago has reduced violent crime by using predictive crime heat maps to aid police efforts [6].

Smart cities go beyond connected infrastructure by engaging and transforming citizens, tourists, and business organizations into an intelligent ecosystem via stimulating innovations [2]. Progressively, city administrations provide data to end-users in an effort to support better decision-making, creating solutions for urban issues, while positively shifting core city operations.

Given the extensive worldwide growth of smart cities, it is intuitive and essential to acknowledge the debilitating and disrupting effects of cyber attacks on these initiatives. For example, an attack that prevented air traffic controllers of a Swedish airport from monitoring aircrafts on their radars [7] could lead to disastrous outcomes. Moreover, the effect is evident in case of an attack on a power grid that left nearly 225,000 people without power in three Ukrainian regions [8]. Additionally, the malicious actions of a ransomware temporarily forced several mission-critical services of the city of Atlanta to become offline, which led to the disruption of utility systems and other key services [9]. Further, Baltimore city not only lost irreplaceable law-enforcement related data but was also subject to a ransomware attack of its emergency service which led to the disruption of the emergency assistance operation [10]. Additionally, distress was caused in Dallas due to the lack of cyber-resilient IoT devices which allowed hackers to disrupt the proper work of road signs [11] and turn on the hurricane sirens in the middle of the night [12]. Such security incidents, as well as many others, impair the trustworthiness of smart cities thus hindering the achievement of their full potential. Therefore, it is imperative to continue to consider and explore the cyber threat landscape of smart cities.

Indeed, new vulnerabilities introduced by the fusion of technological advances, and the complexity, anonymity and severity of cyber threats present new challenges for cyber-related decision making. Notably, the standards and methodologies for cyber security of traditional information technology systems cannot directly be applied to a smart city's ecosystem. This is evident through many recent examples. Hiding behind the cloud, attackers used machine learning algorithms and exploited new evolving technologies such as 5G-enabled IoT to amplify the attack's possibilities [13] and thus the impact. Moreover, the complexity of the interaction between many infrastructure components exponentially increases the impact of cyber attacks. It is inherent that legacy anomaly detection cannot keep up with the rapid agility of an attack, while data-greedy methods

which employ signatures of maliciousness to detect attacks continue to lack behind. Naturally, current research trends points towards harnessing analytics-driven cyber security approaches [14, 15]. For instance, machine learning (including deep learning) is rapidly becoming a key instrument in cyber security as it attempts to effectively and semantically integrate and process the different sources of information.

Protecting smart cities from cyber attacks is a crucial mission for their survival. Although cyber security is among the key challenges for smart cities, the examination of cyber threats is quite a difficult task due to the complexity of a smart city's architecture, its contradictory security requirements, and its use of extensive (vulnerable) software. In this article, we study the availability and sufficiency of data-driven techniques which support cyber situational awareness in the context of smart cities. We uniquely investigate three groups of techniques as classified in Fig. 1; system abstraction, risk and vulnerability assessment, and attack detection methods.

First, to clarify how threats affect the entire ecosystem, we study methods which model dependencies among smart cities' components. Second, we survey risk assessment methods and contextualized threat intelligence (which enable the characterization and anticipation of advanced and coordinated threats) via assessing their possibilities and impacts. Finally, we examine attack detection methods which aid the prompt remediation of threats while contributing to retrospective digital analysis, therefore providing situational awareness and threat prioritization capabilities.

Specifically, we survey, compare and contrast, and discuss each detection method according to the following criteria:

- Theoretical background
- Data input and employed dataset
- Accuracy and performance evaluation metrics
- Scope
- Support of visual aids

In this survey, we will address the following key questions: How does the cyber threat landscape of smart cities look like? What are the available data-driven methods which address the elaboration of situational awareness capabilities? What measurement-based methods can we use to compare the scope and effectiveness of each method? How resilient are the available/current methods against various identified cyber threats?

Along this line of thought, we frame the contributions of this work as follows:

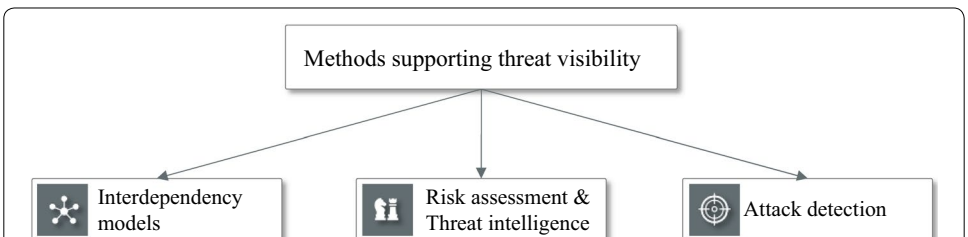

Fig. 1 Scope of the survey

- We classify and enumerate threats targeting smart cities' architectures, link corresponding attacks, and outline the impact of such threats on smart cities' operations.
- We narrate and describe the methods which have been designed to support cyber situational awareness in the context of smart cities. To this end, we consider three groups of methods: (i) interdependency models, (ii) risk and vulnerability assessment, and (iii) attack detection.
- Aiming to put forward a new perspective related to the methods supporting situational awareness, we study their theoretical backgrounds, link available techniques to their corresponding identified threats, and assess the visual support of the corresponding methods.
- To enable and motivate reproducible research methods, we report on fundamental characteristics of the employed datasets that are used for evaluation purposes. Further, we compare and contrast the effectiveness of currently used performance metrics, while suggesting several additional indicators.
- By generating a set of inferences and insights, challenges, and open issues, we put forward potential research directions that can advance cyber situational awareness in the context of smart cities. We also highlight future direction for different research communities, including those of cyber security, data science, and visual analytics.

The rest of this survey is organized as follows. In the next section, we review related surveys and demonstrate the added value of the proposed survey. In "Smart cities: architecture and threat landscape" section, we present the functional architecture of smart cities and indicate several prevalent related cyber threats. In "Methods supporting cyber situational awareness" section, we describe selected detection methods while we correspondingly evaluate them from different perspectives in "Discussion and key findings" section. In "Open questions and future perspective" section, we discuss current research, development and operational challenges while offering possible future research initiatives aiming to advance the currently available analytics for defending against cyber threats. Finally, in "Conclusion" section, we summarize the contributions of each section of this survey.

Related surveys

The worldwide growth of smart cities has induced immense attention from the research community. To highlight the issues and research directions in the context of smart cities' cyber security, a considerable amount of surveys has been published on challenges and recent research trends. These studies provided imperative information on numerous case studies. They also offered information about the functional and technological architecture of smart cities as well as their cyber security challenges, requirements, cyber threats, and respective countermeasures. Although several themes in the surveys overlap, the level of details and vantage points vary from one work to another.

The application of smart technologies and data harnessing methodologies developed numerous solutions to cities' key challenges (including rapid urbanization, increased homelessness, rise in crime, climate change, and more). To illustrate the evolving-nature of smart cities, a number of authors showcased the undertaken technological advancements while providing an overall description [16–18] and a domain-specific

implementation [17, 19–21]. The main focus of these articles was to put illustrate the technological advances and the importance of the interaction between social and technical systems. In fact, this interaction was revealed to be a critical role which addresses the urban challenges and supports innovation and entrepreneurship.

Several other works suggested a functional architecture for smart cities by defining and dissecting the physical, communication, data, and application layers. Here, we noticed a certain level of disagreement in the way the authors compiled these layers. For instance, the data layer is not acknowledged in [22] or is either presented as one [18] or two [21] instances. Surprisingly, none of these architectures mentioned the management layer which is responsible for asset management, service provisioning, and security.

Any discussion of smart urban development should start with a study of infrastructure and enabling technologies. To this end, a plethora of researchers worked on actuators, sensor networks, IoT, Vehicular Ad Hoc Networks (VANETs), Mobile Ad Hoc Networks (MANETs), and access and transmission networks [16–18]. For instance, transmission networks can be used to support better decision-making and resource allocation for smart grids, water and waste management, safety, emergency solutions, and much more by enabling data communication. In fact, the data used for transmission is already collected from the environment by various sensors; we can give for example not only pollution, smoke, motion, and brightness sensors but also cameras, and energy and water meters.

Additionally, a complex interdependent relationship between individual entities in critical infrastructure plays a significant role in the development of smart cities. However, it seems that many available models are unable to capture such complex interdependency to grasp the full array of intentional and accidental threats [23, 24].

Moreover, heterogeneity, limited computational capability, distributed location, and legacy infrastructure raised the attention for cyber security threats and requirements. Unsurprisingly, attacks targeting smart cities' technologies received prevalent attention [16–19, 21, 25–27]. Indeed, requirements and standards are the core factors which determine the essential level of cyber defense and threat prioritization. In fact, strong authentication, secure communication, data protection, monitoring and patching are among the main discussed requirements and countermeasures [16, 17, 20, 21, 25–27] that were envisioned to protect smart cities from adversaries.

Further, given the mounting number of cyber attacks and their impact on the safety of city residents, and on the legacy infrastructure, as well as their corresponding financial concerns require new kinds of situational awareness in order to overcome their resultant implications. In this sense, we perceive a lack of comprehensive studies of the methods supporting cyber situational awareness in the context of smart cities. In fact, combining modeled dependencies of distinct units of smart cities, while exploring and synthesizing contextualized threat intelligence, risk and vulnerability assessment methods, and attack detection techniques would enable adaptive threat modeling by considering the derived knowledge from different sources.

In this context, our work complements the available contributions in twofold. First, we offer a classification of cyber threats and present illustrative attacks and their effects on smart cities' operations. Second, we provide a multidimensional evaluation of available methods that support cyber situational awareness in the context of smart cities and their

enabling technologies. This survey specifically covers methods' classification through analyzing their underlying theoretical models, related data as well as their technological and context-aware scope. It also discusses several evaluation criteria and their visual support (or lack thereof).

Table 1 summarizes and classifies the contributions of the available articles, where the core focus of our work is also highlighted.

Smart cities: architecture and threat landscape

Architecture

The urban infrastructure integrates and ingests rapid technological advances such as digital data and smart analytics, which provides better services to the citizens, improves quality of life, and reduces environmental damages. Although the incorporation of these elements depends on the development strategy and the level of implementation in particular cities, we amalgamate the implementation progress of 10 smart cities in order to model the architecture of smart cities at different operational layers.

The urban infrastructure is comprised of cyber systems integrated into physical components in various environments and includes critical infrastructure like energy, transportation, government, and more. After analysis of diverse case studies [2], we can distinguish 5 tiers of a smart city's architecture, namely, the physical world, enablers, data, applications, and the management layer. For the purpose of completeness, in the illustrated architecture, we consider the stakeholders as providers and consumers.

Figure 2 provides a visual representation of a smart city's architecture along with its 5 layers.

The *physical world* tier encompasses the urban infrastructure and represents buildings, cars, roads, bridges, and streetlights, to name a few.

The *enablers' layer* is comprised of the hardware and communication technologies, which enable the collection of data from the environment and transmit it to the next architectural layer. In fact, the hardware may consist of different types of sensors, devices, or virtual machines. Additionally, various protocols including IEEE 802.15.4, IEEE 802.15.4g, Bluetooth, LoRsa, LoRaWAN empower the sensors for data curation and harvesting so that they could deliver the information to the data layer.

The *data layer* is the heart of smart cities. It consists of an immense volume of unstructured data that should be collected and properly stored to enable open access and the application of numerous algorithms for better decision-making. Actually, artificial intelligence (and broadly learning methods) are among the most popular methods in this layer. The latter is also responsible for placing the data into the visual context to enable the understanding of data significance and to improve the supervisory process. Further, this layer takes care of the data exchange between data owners, service providers, and users by offering open data platforms.

The *applications*' layer represents the variety of complex solutions which smart cities provide to their customers. For instance, data-driven transportation systems address the issues of congestion and pollution, manage parking and public transportation, advance road safety and enhance shipment schedules. To this end, these solutions integrate data from various sources such as geographically-distributed traffic and weather sensors, cameras, and GPS, to apply algorithmic analyzes, while offering optimal paths. Further,

Table 1 A classification of the related surveys

Research area/references	[16]	[17]	[19]	[18]	[23]	[24]	[20]	[21]	[25]	[26]	[27]	This work
Case studies	✓	✓	✓				✓	✓				
Functional architecture				✓			✓	✓				✓
Enabling technologies	✓	✓		✓								
Cybersecurity requirements	✓						✓	✓		✓	✓	
Threat profile												
Exploratory	✓								✓			✓
Enabling technology	✓	✓	✓					✓	✓	✓	✓	✓
Data	✓	✓		✓				✓	✓	✓	✓	✓
Third-party vulnerabilities			✓									✓
Domain-specific	✓		✓								✓	
A classification of threats with examples of attack and impact												✓
Interdependency models					✓	✓						✓
Risk and vulnerability assessment												✓
Attack detection methods												✓
Countermeasures	✓	✓					✓	✓	✓		✓	
Forensics			✓									

The focus of this survey is outlined within the shaded area

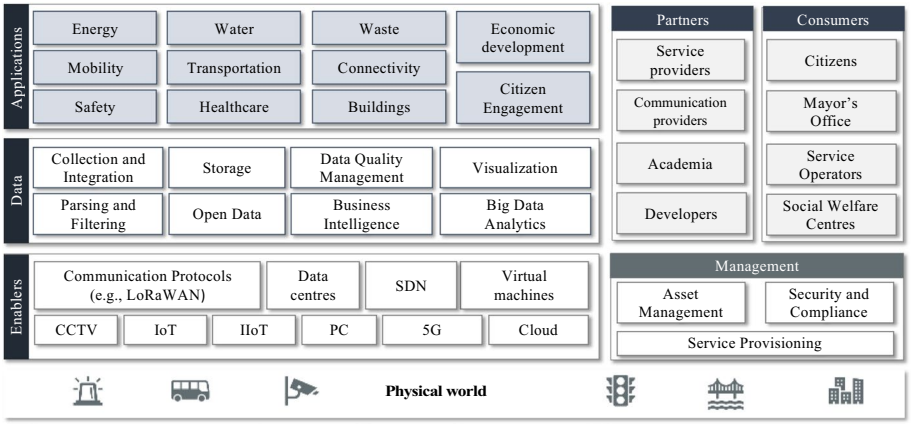

Fig. 2 A high-level architecture of smart cities with different operational layers

the introduction of supported IoT devices overcomes the limitations of conventional monitoring systems. Indeed, sensors collect various environmental measurements such as air pollution levels or water chemical conditions while intelligent platforms correlate obtained data in order to tailor warnings or to avoid ecological disasters. Moreover, the latest developments in power (and micro) grids allows the consumer to monitor electricity consumption in real-time. The latter increases the reliability of power transmission, optimizes the required supply level, and minimizes the consumption cost. Additionally, a data-driven building system processes and responds to surrounding changes by automatically switching air conditioners based on weather predictions or environmental measurements. These are only a few solutions provided by smart cities.

The final layer, namely the *management* tier, addresses service provisioning, asset management, and security. We note that the management of smart cities' solutions can be either centralized or decentralized.

Threat landscape

Such ongoing technological development, however, opened new opportunities for cyber criminals to take advantage of vulnerable cities. Cities around the world continue to become victims of cybercrimes such as hacks, ransomware, utility theft, and loss of control over infrastructure. Predictably, security becomes a very critical challenge for smart cities [28]. Indeed, there are several reasons why the implementation of cyber security in smart cities' settings is significantly tricky, including heterogeneity and geographical distribution of the infrastructure, difficulties of patching, often the limited computational capabilities of distributed devices, usage of legacy equipment, difficulties with security and privacy measurements, and inconsistencies within security requirements.

While the threat landscape depends on the development level and on the architecture of a particular smart city, in this section, we discuss prevalent threats targeting smart cities. Please note that we exclude from this survey threats such as hardware failure, software and human errors, and electrical interruption.

Additionally, the architecture of smart cities is vulnerable to traditional computer viruses, remote breaks, eavesdropping, software hijacking, injection of malicious

contents and/or malformed requests, memory exploits, access to sensitive information, and data misuse. Moreover, the connection with the physical environment, the introduction of IoT devices, and the emergence of new communication protocols, as well as employing machine learning algorithms, induce new security threats and amplify the impact of traditional exposures. Motivated by the aforementioned information, we define the following four classes of threats as depicted in Fig. 3; *exploratory threats*, *infrastructure sabotage*, *data manipulation*, and *third-party vulnerabilities.* Indeed, the immense danger of these threats is rooted in the elevated interdependencies between different components of smart cities.

First, the class "*exploratory threat*" consists of the threats aiming to enumerate resources and credentials. Second, "*infrastructure sabotage*" represents threats aiming to destroy or gain control of smart city infrastructure by deploying malware and reprogramming or overwhelming core resources. Third, the class "*data manipulation*" consists of threats which endeavor to undermine data confidentiality and integrity, as well as destabilize machine learning algorithms. Finally, "*third-party vulnerabilities*" refers to threats targeting service providers which produce a profound effect on smart cities' activities and security.

As noted in the previous section, smart technologies are widely used in critical infrastructure sectors in order to provide valuable services to consumers. Therefore, it is imperative to realize how the identified threats jeopardize the business continuity of smart cities. However, lack of visibility into real-time data regarding incidents makes it difficult to quantify such impact.

Exploratory threats

The first step of any attack is the exploratory step, during which an attacker gathers valuable intelligence about the target. It starts by infiltrating target assets (e.g., exposed systems, devices) in order to explore possible points of entry of the system. Further, as summarized in Fig. 4, an adversary would almost always employs various methods to enumerate the deployed infrastructure.

Resource infiltration To index vulnerable IoT devices that have been deployed in smart cities, the attacker can perform active Internet-scale probing [29] or use the search engine Shodan [30] to discover easy targets with default credentials [31]. Besides being a first step of any cyber attack, network scanning significantly degrades network performance, thus also slowing down response times of users' requests [32].

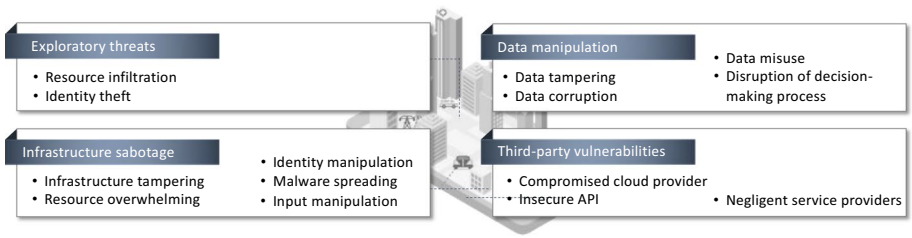

Fig. 3 Threat landscape of smart cities

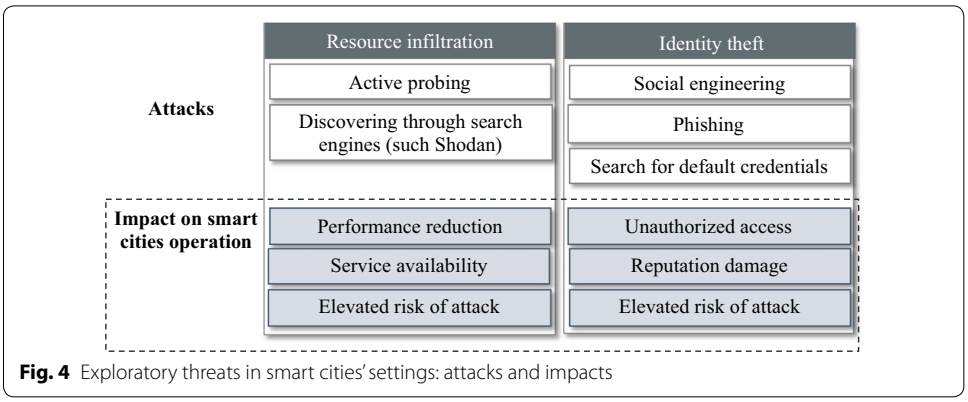

Fig. 4 Exploratory threats in smart cities' settings: attacks and impacts

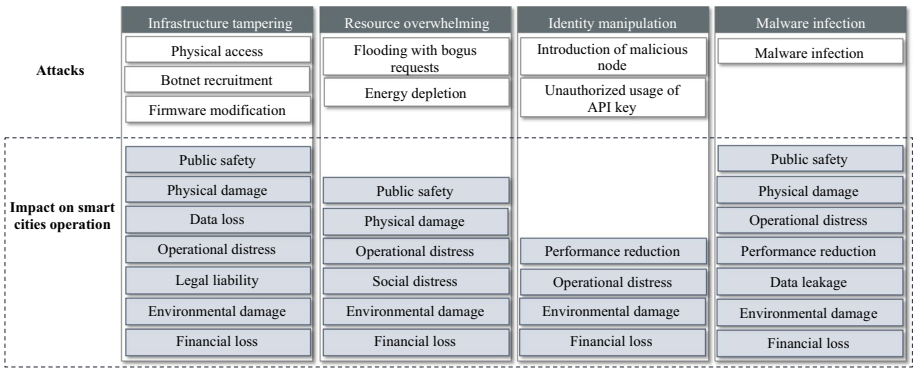

Fig. 5 Infrastructure sabotage in smart cities' settings: attacks and impacts

Identity theft involves obtaining the victim's identity to gain privileged access to the system or data, take control over the infrastructure, and conduct various attacks. The threat comes in different forms; the credentials could be extracted from hashes, social engineering, and phishing, or by leveraging weak credentials of IoT devices. The latter is a specific threat to IoT-based smart cities since their solutions rely on data collected by IoT sensors. Additionally, the costs of credentials theft come in the form of unauthorized access to smart cities' resources. Indeed, the loss of control over the infrastructure and data promotes the risk of reputational damages.

Infrastructure sabotage

One of the attacker's motivation is to gain illegitimate control over the infrastructure through tampering, manipulating, reprogramming, or overwhelming the resources. Figure 5 illustrates attacks associated with this category and their impact on smart cities' operations.

Infrastructure tampering This threat could be manifested in two ways; directly or remotely. Since a large number of actuators and sensors operate in an unattended fashion with either no or limited tamper-resistance policies and methodologies, an attacker could take advantage of a physical access to a device intending to cause significant

damage, alter its services or obtain unlimited access to data stored on its memory. In fact, compromised actuators which control the physical infrastructure (e.g., heating, switching elements, etc.) can provoke damage to physical objects and threaten public safety. Further, the attacker can use the smart cities' infrastructure to recruit it into a botnet, causing both direct and indirect adverse effects. The former implies losing control over the infrastructure and putting the city's critical functions at risk. Indirectly, the bots can be used by the attacker to launch Distributed Denial of Service (DDoS) attacks, harvest information from the network [33], mine cryptocurrency [34], or distribute malware [35], to name a few. In fact, the indirect impact can cause system's performance degradation, in addition to legal and compliance liability. Moreover, an attacker can also tamper a device by exploiting firmware vulnerabilities [36]. Indeed, firmware modification is rendered by maliciously altering the firmware which induces a functional disruption of the targeted device.

Resource overwhelming The adversary here is seeking to disrupt the service by preventing access to that service. To that end, the adversary floods the target with excessive requests. As a result, the service cannot process all requests and therefore legitimate users cannot gain access. Additionally, this is a threat for APIs since the service does not limit the number of received requests. It is also true for IoT-based smart cities, given the limited computational capabilities of the IoT devices. For instance, in Finland, a building management system was flooded with bogus requests forcing heating devices to become offline [37]. Further, strict safety regulations and radio propagation limitations prevent embedded devices from efficient energy harvesting [38] which allows the attackers to drain energy from the smart cities' infrastructure. Additionally, poor software development practices can significantly increase energy consumption [39] and lead to the disruption of a city's operation.

Identity manipulation This threat could originate into two different ways; by either introducing fake sensors or by using unauthorized API keys. Additionally, introducing malicious nodes into a network leads to the degradation of a network's performance.

Malware spreading An attacker spreads malware with the intent to infect smart sensors, IoT devices, or data servers. To this end, he attempts to change their functions or leak sensitive data. For instance, sending fake overload status from a wide range of smart meters could force several segments of the grid services to become offline.

Data manipulation

Data manipulation threats manifest in four different ways; data tampering, data corruption, data misuse, and decision-making process disruption. Figure 6 illustrates the attacks associated with this category and their impacts on smart cities' operations.

Data tampering Recently, widespread ransomware attacks paralyzed numerous smart cities. Ransomware is malicious software that locks hardware or encrypts data files until a monetary ransom is paid typically through cryptocurrencies. Due to the highly connected nature of smart cities, this type of threat has a massive impact on city operations. For example, the malware SamSam held hostage the services provided by numerous cities. This example shows how data tampering can sabotage data-dependent functions and results in massive data loss [9]. Additionally, if an adversary attacks vital sectors such as healthcare [40], the inability to access patient data could cost human lives. Moreover, we

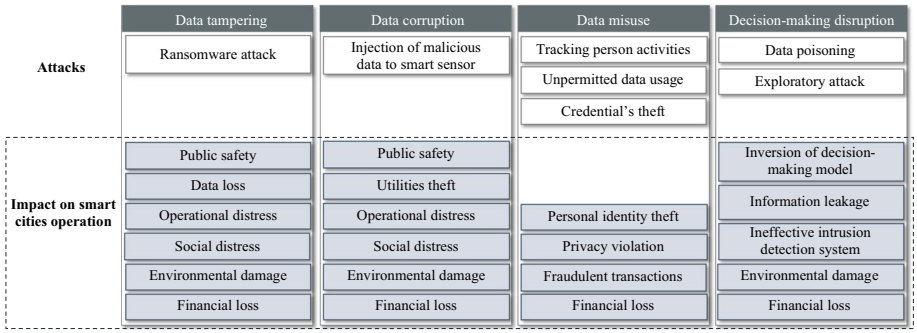

Fig. 6 Data manipulation threats in smart cities' settings: attacks and impacts

can quantify data tampering attacks in smart cities by measuring their financial loss. For instance, paid ransom, price of data recovery, and operation loss (e.g., free bus rides [41]) are several examples which assess the financial loss caused by such attacks.

Data corruption Even though injecting malicious data in smart sensors seem minimal [42], it could cause a dramatic economic effect or cost human lives [43]. For instance, smart meters can be used to steal energy from municipalities [44]. Additionally, emergency alerts can be used to create havoc [11]. Further, these malicious inputs can be crafted in such a way that they force machine learning models to make false predictions and cause instability in the city's operations.

Data misuse Smart cities' infrastructure generates vast amounts of information. Indeed, this information is collected from myriad of sensors and from citizens (when applicable, with their permission). Additionally, the information collected can be used to take advantage of personal information for various reasons. Indeed, an attacker could want to track a person's activity through sensors or surveillance cameras, sniff the communication or leverage weak web access to steal a person's credentials which could later on be used for fraudulent transactions. However, data misuse does not necessarily arise when the collected data is used for unpermitted purposes [45].

Disruption of decision-making process Since most smart cities implement machine learning algorithms as a decision engine, we should consider the reliability of such algorithms. In fact, attackers seek a disruption of the decision-making process by data poisoning or by instantiating an exploratory attack. In the case of data poisoning, the decision engine is compromised by injecting carefully designed adversarial samples to the training dataset with the aim of compromising the learning process. Indeed, the significance of the exposure occurs from the difficulties in recognizing the correctness of the produced output due to the complexity of machine learning algorithms. Additionally, the trustworthiness of smart cities' analytics will continue to suffer until the robustness of such algorithms can be confirmed. This issue is equally relevant for the trustworthiness of intrusion detection systems backed by machine learning algorithms [46]. In case of exploratory attacks, the adversary tries to gain information by probing the learner. In fact, it operates by injecting samples that are designed to bypass the learner during the testing phase with the intention of model inversion or information inference.

Third party vulnerabilities

Smart cities' administration typically decides to collaborate with the private sector to augment budget and skills limitations while promoting innovations and economic development. In fact, the private sector designs and supplies enabling technology, builds infrastructure, collects and processes data and develops software supporting decision making. This strategic collaboration, however, brings a new edge for cyber threats. Figure 7 illustrates such threats/vulnerabilities, associated attacks, and the impact of these attacks on smart cities.

Compromised cloud provider Cloud computing brings a new frontier for the developers by proposing an infrastructure, platforms and software over the Internet. Indeed, cloud services are an attractive option for ever-growing smart cities because of the low-level of initial investments, their scalability and continuous availability. However, the features of cloud computing, such as multi-tenancy and virtualization, can lead to the leakage of private information and unauthorized access. Representative examples of such attacks include those which hijack cloud accounts and exploit system vulnerabilities [33].

Insecure Application Programming Interfaces (API) is a vulnerability that opens the doors to cloud applications and web services. In fact, API allows the users to customize their experience and receive access to many web services, including authentication and access control. Actually, APIs are designed to share information. Therefore, the impact of API breaches depends on the service and shared information. For instance, in the city of Los Angeles, a clear text API key allowed a hacker to use paid Google services [47].

Negligent service provider Some vendors introduce new cyber vulnerabilities. They can intentionally or unintentionally leave backdoors that allows attackers to access devices or software. They can further deploy IoT devices without patching capabilities. Although smart cities often operate unpatched, vulnerable products, some vendors are resistant to acknowledge security holes in their products, which endanger smart city service operators and subsequently the citizens.

Methods supporting cyber situational awareness

Even though the previously described threats against smart cities continuously evolve, many advanced methods continue to be developed to support cyber threats' visibility. In this section, we describe the selected methods followed by their categorization in "Discussion and key findings" section.

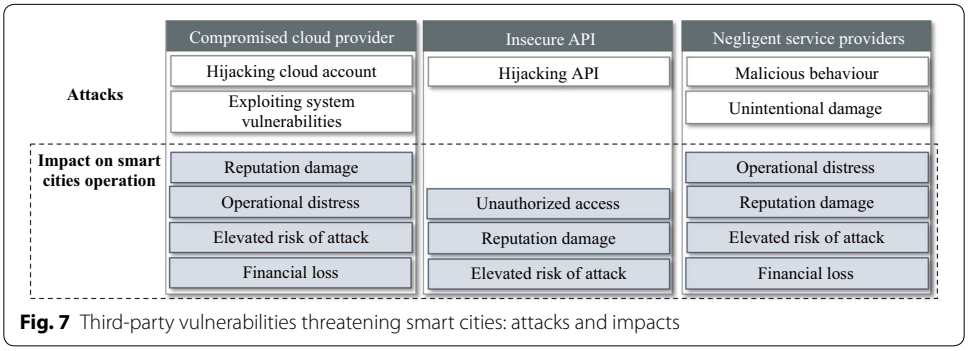

Fig. 7 Third-party vulnerabilities threatening smart cities: attacks and impacts

Interdependency models

Heterogeneous communication protocols and shared infrastructure connect various embedded systems to make cities more effective. Additionally, different service providers exchange information and resources to support the sustainable operation of a smart city. This high interdependence introduces a large number of possible attacks and vulnerabilities that directly relate to the severity of the threat and have a multiplicative effect on the prioritization of mitigation. Indeed, a threat that results in the loss of one service or infrastructure can potentially impact other services as they use each other's resources. Moreover, identifying these vulnerabilities and their impact is challenging because of the high complexity of the connection among different infrastructure. Further, each smart city's component has a variety of security requirements which introduces additional challenges.

Any disruption in smart cities' systems would have an impact on its effective operation as well as the safety and well-being of its citizens. Additionally, a formal dependency model of the smart city's elements would uncover insights into fundamental characteristics of the system's topology and could be instrumental in developing its security profile, assessing the cumulative impact of cyber threats, and estimating the effect of countermeasures. While the discussed dependency models do not consider cyber threats, understanding the connection between different domains affects threat prioritization and mitigation.

To this end, Laugé et al. [48] demonstrated how a failure in one service could affect other domains. In this context, the researchers conducted a series of interviews with experts and quantified the magnitude of the adverse effect on dependent services such as energy and connectivity. The results, which include characterization of the time dimension to dynamically study the impact, enabled a deep understanding of direct and higher-order dependencies to prioritize mitigation.

Further, König et al. [49] proposed a framework to represent the effect of adverse events in highly coupled critical infrastructures (CI). The approach modeled the dependencies between infrastructures as a directed graph. In fact, each CI is denoted as a single vertex, while the edges symbolize the reliance on the others CIs' resources. Each edge is then assigned to a class $c \in \{1, 2, \ldots C\}$ which represents a fixed type of inner or mutual dependency. Additionally, these dependencies are assessed using a Markov chain and by leveraging interviews with experts. Moreover, the visualization of dependencies illustrated how the limitations in one CI affect dependent CIs and how this impact changes over time.

To identify the minimum subset of critical infrastructure nodes and select the most rewarding mitigation priorities, Stergiopoulos et al. [50] input a dependency risk graph into their model and define the correlation between centrality metrics and high impact nodes. Further, the authors used centrality metrics to develop and test various risk mitigation strategies that maximize risk reduction. The results demonstrated that centrality measures could characterize critical infrastructure nodes that significantly affect the overall risk in a dependency risk graph.

In an alternative work, Stergiopoulos et al. [51] modeled dependencies among infrastructures as a graph $G = (N, E)$, where N is a set of nodes representing infrastructures or components, and E is a set of edges that symbolize dependencies. In fact, an edge from node CI_i to node CI_{ij}, i.e., $CI_i \rightarrow CI_i$, denotes a risk relation that is derived from

the dependence of infrastructure CI_j on a service provided by infrastructure CI_i. This relation is quantified using the impact $I_{i,j}$ and the likelihood $L_{i,j}$ that a disruption will be realized. Additionally, the cascading resulting risk is represented as a numerical value of each edge. The growth level is then precomputed and is passed to a fuzzy ranking system that provides realistic assessments of the evolution of potential failures.

One of the goals of Beccuti et al. [52] was to investigate the consequences of a malfunctioning communication system when the power grid experienced a failure. To this end, the authors modeled and simulated the electrical state of the Electrical Power System (EPS) using a Stochastic Activity Network (SAN). In contrast, a Denial of Service (DoS) attack was modeled using Stochastic Well-formed Nets (SWN). The researchers investigated how these two models can be integrated to characterize the DoS attack impact. While the approach is focused on specific scenarios, the executed analysis illustrated that the user satisfaction of a power line can differ significantly depending on the severity and progression of the DoS attack.

In a different work, Bloomfield et al. [53] centered their study on how the strength of dependencies between power and telecommunication networks affects various measures of risk and uncertainty. The approach begins with a high-level of abstraction aiming to identify dependencies between the components of CIs which is then followed by a detailed service behavior model. Further, the authors employed probabilistic models to come up with various measures for risk assessment, e.g. the likelihood of cascade failure under a given set of assumptions.

Netkachov et al. [54] used stochastic modeling of an industrial control system and studied the effect of both accidental failure and cyber attacks. In fact, the researchers used a stochastic state machine to model the behavior of the adversary while the dependencies between the elements are modeled using a deterministic or a probabilistic approach. The study of the employed approach unveiled the most critical elements of the network and a high correlation between the impact and the capability of the attackers.

Further, Johansen et al. [55] proposed to model the interdependencies by using a Bayesian network and a minimum link set (MLS) formulation to create the network model. The latter represented a set of functioning components required from the system to function. Moreover, the authors distinguished three types of dependencies; service provision, geographic, and access to repair interdependencies. This dependency relationship was then defined by the joint probability distribution of the components. Regardless of parent choice, the entire system is defined using joint probabilities divided by the marginal chances of failure. By applying their framework on a real system and given the complex interdependencies, the researchers quantified the cascading effect of an individual component's performance on the entire network performance.

Additionally, Heracleous et al. [56] proposed a dependency modeling method that supports the investigation of the cascading effect, performs vulnerability analysis, and plans maintenance strategies. The authors demonstrated how an open hybrid automata allows modeling individual subsystems and composing them together to create more complex and detailed systems with the aim to capture different types of dependencies. By connecting six open automata models that represent various components of CIs, the authors ran simulations to study the effect of the malfunctioning of one infrastructure on other elements, perform vulnerability analysis, and offer a maintenance plan.

In another work, Ferdowsi et al. [57] analyzed the problem of allocating security resources over the various components of interdependent cyber-physical systems (CPS) in order to protect the entire ecosystem against cyber attacks. Indeed, the authors formulated a Colonel Blotto game where the attacker seeks to allocate its resources with the intention of compromising the CPS. At the same time, the defender chooses how to prioritize the defense against potential attacks. The reported result illustrated the correlation between the attacker's knowledge of interdependencies and the defense's success.

Risk assessment and threat intelligence

The growing number and scale of cyber threats demand proactive decisions for the development of ample cyber security capabilities. In fact, the core challenges for cyber-related decision making are the uncertainty of cyber threats and their severity, and the technological advances that introduce new vulnerabilities. Given the heterogeneity of IoT devices, a myriad of vulnerabilities requires patching and monitoring. Therefore, it is imperative to set the priority to secure critical weaknesses and allocate time and budget effectively. Contextualized cyber threat intelligence capabilities complement the risk assessment objective by helping discover unknown incidents, attack trends while assessing and comprehending their impacts.

In the context of risk assessment, Li et al. [58] estimated cyber security risk in traffic light systems. The authors first employed a game-theoretic framework to determine the worst-case traffic management performance under attack. The metric is then used to determine the severity of a particular attack as $S_i = P_0 - P_i^*$, where P_0 represents a system performance that is not under an attack and P_i^* represents a system performance under an attack. The researchers then determined a cyber security risk of a traffic light system under a certain traffic network condition by calculating it as $R = \sum_{i \in C} L_i * S_i$. Further, a cyber-risk mitigation framework is formulated using subjective decision rule known as a minimax-regret criterion. Here, the regret is defined as the risk under a specific traffic condition with no countermeasures employed. Additionally, the ranked countermeasures manage to minimize the worst-case regret.

Kelarestaghi et al. [59] conducted a vulnerability-oriented risk assessment by employing a National Institute of Standards and Technology (NIST) risk model. The authors synthesized real-world misdemeanors and research publications that study the attacks against in-vehicle network vulnerabilities in order to quantify the potential impact of the exploitations. Safety, operational, and security issues were then mapped into a visual matrix to facilitate risk prioritization. Moreover, an empirical study unveiled the severe impact of cyber attacks on the safety, security, and operation of the vehicle.

In an alternative work, Kotzanikolaou et al. [60] assessed a possible cascading effect of a single incident on multiple CIs. In fact, the approach models the connections between infrastructure as a graph where the edges represent the dependencies under regular operation. Additionally, the method does not differentiate the risks but uses the impact of adverse effects as a result of a risk assessment for each infrastructure.

It is hard to overestimate the importance of IoT in a smart city's ecosystem. Given the diversity of IoT devices, the vulnerabilities of the entire system are countless [61]. Sicari et al. [62] proposed a general-purpose risk assessment methodology in

the context of IoT deployment. The framework first identifies the model's components and forms an attack tree with the nodes representing a different way of attacks and the leaves symbolizing the vulnerabilities v_i. Indeed, each vulnerability is associated with an exploitability level E_i. The latter indicates a measure of how probable the v_i is exploited to perform the attack. In the next step, the framework models a graph to depict the dependencies d_i among v_i. The exploitability level is then assigned to each edge of the graph and is updated according to the formula $E_{i+1} = \max(E_0(v_i), \min(E(d_i), E_i(E_i)))$, which indicates the risk of exploitation. Moreover, the approach enables scalability in terms of effortless adding or removing components from the framework.

Further, Wang et al. [63] proposed a vulnerability assessment method rooted in an attribute attack graph. In fact, the model takes a network topology, the vulnerabilities, and an attack graph to generate an optimal attack map. It further calculates max loss from the exploitation by using a score from the Common Vulnerability Scoring System (CVSS) [64]. Finally, the model employs an augmented path algorithm to suggest an attack priority order and determines the weakest link in the system to prioritize their monitoring and security.

In a complementary work, Radanliev et al. [65] proposed an economic impact assessment framework for IoT. The authors adopted the Cyber Value at Risk model to measure the maximum possible loss over a given time period and the MicroMort model to predict uncertainty through units of mortality risk.

Nazeeruddin [66] leveraged Markov's decision process in order to model the security of smart cities at a high level of abstraction. The model considers the system components and their types (e.g., sensor, actuator, etc.), the cyber attack against each element, the vulnerabilities with the exploitation probabilities that are extracted from the CVSS database, and the human involvement at the last level of defense. In case the attack successfully passed two levels of defense mechanisms, the model generates an alert for review by security analysts for further investigation. The authors demonstrated that the model could easily be adjusted with vulnerabilities to recalculate the risk level.

Shivraj et al. [67] offered a generic risk assessment framework for IoT systems. The authors described information flow across the different components as a weighted directed acyclic graph G(V, E). The edge E between nodes V indicates a dependency of one node to another. Indeed, one node can be connected to multiple ones, producing numerous connections. Additionally, the value of the edge weight is directly linked to the impact of the attacks. Moreover, various attacks are modeled through attack trees, while their propagation is represented using a bipartite graph. The latter allows capturing nested attacks (e.g., through spoofing an attack on a node; tampering, spoofing and denial of service attacks can be carried out on its parent node). The authors demonstrated the risk computation based on a simulated system of a connected car.

Mohsin et al. [68] proposed a probabilistic model aiming to automatically assess the likelihood of a threat realization in various IoT system configurations. At the very first stage, the framework leverages a Markov model to represent the system's architecture, security threats, and attackers' capabilities to predict the likelihood of an attack and

suggest a secure configuration. Additionally, the framework addresses both concurrent and sequential elements of the system by assigning the synchronization labels for modeling concurrency, flags and counters for the subsequent flow. Moreover, the framework, dubbed as *IoTRiskAnalyzer*, answers the question of what the best possible configuration for a security requirement is, and how promising it is to enable the diagnostic of a cyber security posture.

One of the core goals of advanced threat detection is to determine the potential progress of the discovered malicious event through the ecosystem. In this context, Falco et al. [69] designed a method for automatic identification of attack strategies that can be used to compromise a CCTV network. The approach combines several established frameworks to address the full lifecycle of the attack. Additionally, a Lockheed Martin's cyber kill chain is used to define the sequential phases. Moreover, the Open Web Application Security Project (OWASP) allowed identifying attack surface areas. Further, a MITRE's Common Attack Pattern Enumeration and Classifications (CAPEC), along with Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK) framework defined the required actions to conduct the attack. Finally, Kali Linux tools and known exploit tactics by MITRE's ATT&CK Matrix execute the actions. The result, compared with the manually generated attack tree, demonstrated considerably greater depth and information granularity than the manual tree because it moves through each phase of the Cyber Kill Chain.

Angelini et al. [70] associated network topology and geography with the resultant impact using a visualization based on areas of corruption. This method was used in order to concentrate the attention on the most harmful risk of cyber incidents. In fact, the method's architecture is comprised of several components, including knowledge base generation, attack, risk, and response modeling. First, the model defines business processes of the power distribution system, and then assigns the mission priority and the cyber events that can adversely affect the business process. For visualization purposes, the authors clustered dense areas of network nodes and employed the Voronoi diagram to effectively spot the geographical placement. The reported results highlighted the subnetwork which could cause mission degradation if compromised.

To analyze the degree of exploitation, Wang et al. [71] measured smart cities threat factors by combining more than 200 gathered features based on a Hardware, intelligence, Software, Policies and Operation (HiSPO) approach [72]. After assigning a weight $w_i = 1/\sum_i(r_i)$ to each threat, the threat factor was calculated as $t = 0.5 * \sum w_i * (t_i + \delta) + 0.001 * (C_B + C_T + C_E) + 0.02 * f_{TI}$, where C_B, C_T, C_E are base, temporal, and environmental scores in CVSS, respectively. Additionally, an adjusted weight for a threat is denoted as δ, while f_{TI} symbolizes a threat intelligence value. Moreover, the final report produced threat factors that were calculated before mitigation and after the assessment and mitigation period. Further, it showed that the proposed methodology can considerably minimize the risks for smart cities.

In an alternative work, Bou-Harb et al. [73] prototyped a IoT cyber threat intelligence platform for inferring and disclosing Internet-scale compromised IoT devices. To this end, the authors amalgamated the results from passive and active measurements of Internet-wide network traffic analysis. In fact, through an authenticated platform, they disclosed raw data related to numerous compromised IoT devices in diverse sectors,

including critical infrastructure. Indeed, the platform estimates the indicators of a highly exploited hosting environment to provide early warnings regarding such exploitations and leverage visual dashboards in order to facilitate threat exploration and prioritization.

Further, honeypots trap an adversary by intentionally creating security vulnerabilities in specific technologies. These devices (or software) record malicious activities so that attack vectors and patterns can be further investigated. Given that ZigBee-based IoT devices are actively used in smart cities settings [74], the honeypot that simulates a ZigBee gateway proposed by Dowling et al. [75] is instrumental to explore attacks against smart cities. After 3-month of monitoring the activity that has targeted the ZigBee gateway, the researchers reported 6 types of executed attacks. These include dictionary and brute force attacks, scans, botnets and a number of other independent events. The authors also reported that dictionary attacks represented nearly 94% of all attacks.

Attack detection methods

Data-driven threat assessment, though extremely valuable and insightful, cannot capture all possible threat capabilities. To this end, a retrospective incident analysis captures several threat attributes and system characteristics, which allows the measurement of the effectiveness of the implemented defense mechanisms. Indeed, scientific efforts towards the development of compelling techniques for the detection of threats and malicious events have been studied for decades, yielding a plethora of inference methods. A recent trend continues to converge towards machine learning techniques, which addresses the problem of recognizing malicious patterns in (network) data flows/traffic to infer anomalies.

In this vein, the main goal of the work conducted by Oza et al. [76] is to detect replay attacks—a subset of false data injection attacks—in an effort to secure traffic lights. Indeed, such attacks minimize the efficiency of traffic management systems, and potentially can introduce life-threatening situations. To this end, the authors simulated a replay attack and studied existing detection mechanisms. They identified several shortcomings in these mechanisms and offered a threshold-based method for detecting an attack. Additionally, the authors determined a threshold by analyzing the occupancy's sensors' readings with and without attacks. The detection algorithm observes the occupancy's sensor's data over time and alarms the operator if the change is above a defined threshold.

To detect energy theft, He et al. [77] attempted to identify potential malicious injections in the context of a power grid. The authors proposed a real-time scheme for capturing the behavioral features of false data injection attacks. Indeed, the architecture of the solution consists of a State Vector Estimator (SVE) and a Conditional Deep Belief Network (CDBN). The latter consists of a Conditional Gaussian–Bernoulli RBM method at the first hidden layer and a conventional RBM technique at all remaining hidden layers. Additionally, the CDBN is responsible for the extraction of high-dimensional temporal features. Moreover, the SVE evaluates the quality of the measurement data by calculating the l_2-norm of residual measurement and compares the calculation result η with the predetermined threshold τ. Further, when $\eta > \tau$, the measurement is considered to be compromised.

The infrastructure of smart cities, particularly those aspects dealing with IoT devices, can be infected by malware or recruited into botnets for conducting DDoS attacks and other coordinated events. To this end, Azmoodeh et al. [78] applied a convolution network to the vector representation of Operations Codes (OpCodes) to detect IoT malware. The model first generates the graph of OpCodes and then converts it to eigenspace (i.e., eigenvector and eigenvalue) in order to pass it as an input to a convolutional network.

Further, Dovom et al. [79] proposed a malware classification technique rooted in fuzzy and fast fuzzy pattern tree that were applied to a vector representation of OpCodes sequences. In a nutshell, a fuzzy pattern classifier is a collection of fuzzy pattern trees $PT = \{PT_i | i = 1, \ldots k\}$, and each PT_i is a pattern tree associated with class $y_i \in \{malware, benign\}$. The tree that produces a higher score $\hat{y} = argmax(PT_i(x))f$ *or* $y_i \in \{malware, benign\}$ is then used to assign the class. In fact, the authors leveraged a class-wise information gain to select the most beneficial features for flow graph generation. Additionally, the proposed method outperformed SVM, KNN, Random Forest, and Decision Tree classifiers. Moreover, the proposed method demonstrated a general potential in interacting with noise and ambiguity, making it a considerable solution for deployment at the edge of a network.

Malicious behaviors of recruited IoT devices (into botnets) can be detected in different stages of the attacks. Along this line of thought, Kumar et al. [80] endeavored to detect individual bots before an actual attack, i.e. during the scanning phase. Indeed, they analyzed network activities for early detection of individual bots. Towards this, several machine learning algorithms, such as Random Forest, KNN, and Gaussian Naive Bayes were used to label the network traffic that demonstrates a behavior similar to an IoT-botnet behavior. To increase the performance of the method, the authors operated on an aggregate traffic in order to detect an IoT access gateway-level. This method was proved to be faster and reduced the memory space required.

Alternatively, since some attackers made successful attempts to avoid detection, it is crucial to be able to detect the infections in later stages of the attack. To this end, Meidan et al. [81] proposed N-BaIoT, a network-based approach which detects compromised IoT devices that are used to launch attacks. The approach extracts statistical features that capture the behavior of the network and uses deep autoencoders (DAE) in order to detect anomalous network traffic generated by compromised IoT devices. The method was proven to be able to detect previously unseen botnets with low rates of false alarms, which is crucial for resource allocation.

In an alternative work, Alazab et al. [82] proposed a detection technique which semantically discriminates botnets and verifies the behavioral legitimacy of numerous smart city's IoT-based applications. Indeed, the authors leveraged the domain name system's (DNS) services to build-upon a framework which initially visualizes DNS features (such as domain name length, domain name entropy, and domain name n-gram). Consequently, the method estimates a similarity score and compares it with a predefined threshold. The domain names that did not pass the threshold are labeled as spoofed. Additionally, a cost-sensitive deep learning algorithm analyzes other domains. Here, the results are also visualized for the administrator for easy of digestion.

Alternatively, Raza et al. [83] proposed a method to detect attacks inside the 6LoWPAN network protocol, which is actively used in smart lighting solutions. By observing a network topology, the framework's modules grasp inconsistencies in node communications and detect attacks. First, the approach gathers information about the network to reconstruct a Destination-Oriented Directed Acyclic Graph (DODAG). Then, it infuses the node's parent and neighbor information into the graph. An algorithm which analyzes consistency in a network graph carries the detection of false data injection and routing attacks. In an extended version [84], the authors leveraged Expected Transmissions (ETX) metrics, which are measured by sending periodical probe packets between the participating neighbors.

By modeling non-linear correlation among multiple time series, Li et al. [85] designed an unsupervised GAN-based anomaly detection (GAN-AD) method for inferring attacks in multi-process CPS with various network sensors and actuators. The proposed GAN employed Long Short Term-Recurrent Neural Networks (LSTM-RNN) for both the generator and discriminator and calculated scores to indicate the level of abnormality in the time series. In fact, when tested on the CPS dataset from the Secure Water Treatment Testbed (SwaT), the model demonstrated that it outperformed existing unsupervised detection methods.

Alternatively, to detect crypto-ransomware in IoT networks, Azmoodeh et al. [86] classified power usage patterns on IoT nodes and discriminated ransomware-infected nodes. At the first stage, the methodology recorded a sequence of energy usage for each process of the targeted devices, followed by a calculation of the distance that measures an optimal alignment between two time-dependent sequences known as Dynamic Time Warping (DTW). Finally, the authors employed three classifiers, namely Neural Network, SVM, and KNN. In combination with Dynamic time warping, KNN outperformed other classifiers and demonstrated remarkable performance (94.27%) in detecting ransomware in the IoT nodes.

One of the biggest cyber security concerns directly refers to the inability of machine learning methods to combat adversarial attacks. Indeed, the proactive data-driven defense methods that aim to cope with the attack against machine learning algorithms propose to sanitize the training and testing data by detecting the adversarial injection. For instance, Baracaldo et al. [87] leveraged provenance data, which consists of metadata describing the origin and lineage of each data point, in order to identify malicious manipulation of the training data. Additionally, the authors pinpointed generated poisoned data, formed provenance data. Moreover, the validation unveiled that employing this method as a filter during the training phase significantly improves classification performance.

Further, the framework proposed by Laishram and Phoha [88] clusters the feature space of the input and filters out suspicious data points. The method calculates an average distance of each data point from the other points in the same cluster. It then considers a class label as an additional feature with a proper weight. Additionally, the data points with a confidence level less than 95% are removed from the training data to achieve input purity. Moreover, empirical experiments demonstrated remarkable accuracy improvement of the SVM classifier.

Discussion and key findings

In this section, we evaluate the reviewed methods and extract findings and insights from the conducted literature review. First, we distinguish and briefly describe employed theoretical background by pointing the category of models that use it. We further link the models to the identified threats to analyze the covered scope. Additionally, we take a closer look at the plausible visual support that could be provided by the reviewed works. Lastly, we establish a benchmark for comparison by examining the main characteristics of the underlying data and by reporting several evaluation metrics.

Theoretical background

We distinguish two classes of models based on their theoretical background, namely (i) *machine* learning *and data mining*, *and* (ii) *knowledge-based models*. The first class consists of methods which derive complex pattern-matching capabilities from the training data; hence it includes a learning stage. The second class consists of methods which require creating a knowledge base that reflects a system or a security profile. These classes along with their subclasses are shown in Fig. 8.

Machine learning and data mining methods

Deep autoencoder (DAE) is a feed-forward multi-layer neural network that is trained to compress and reconstruct input data with a minimal difference between input and output [89]. $\overline{X} = D(E(X))$, where X and $\overline{X}$ are input and output, respectively, E is an encoder from the input to the hidden layer, and D is a decoder from the hidden layer to the output. Indeed, a DAE is designed to prioritize the features of X that should be copied to $\overline{X}$. Therefore, it learns important properties of the underlying data. Additionally, the goal of DAE can be formalized as the following optimization problem: $\min_{D,E} \|X - D(E(X))\|$.

Deep Belief Networks (DBNs) [90] consist of multiple layers of stochastic and latent variables and can be regarded as a special form of the Bayesian probabilistic generative model.

Convolutional Network is a neural network that consists of convolutional and subsampling layers followed by fully connected layers. In fact, the convolutional layer has k kernels that act as a feature detector.

Fig. 8 The categories of theoretical methods. Labels: —employed for dependency modeling, —employed for risk assessment, —employed for attack detection

Fast fuzzy pattern tree [91] is a tree-like structure in which the inner nodes are fuzzy logic arithmetic operators and the leaf nodes are associated with fuzzy predicates on input attribute.

Generative Adversarial Networks (GANs) [92] is a generative and discriminative deep learning architecture that consists of two competing neural network models, namely generator (G) and discriminator (D). As a first step, the generator receives noise z to learn a distribution p_z. Based on perceived distribution, the generator G produces data samples and passes them to the discriminator D. Then, the discriminator uses the Jensen–Shannon divergence to determine the distribution between real and fake data, and back-propagate the probability of data authenticity to G. Additionally, the generator subsequently adapts its parameters based on received gradient information and passes new samples to D. The goals of the generator and the discriminator can be formalized as the following minimax game with value function $V(D,G) : \min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log(D(x))] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]$, where p_{data} is the data distribution and p_z is the prior distribution of the generative network.

Gaussian Naive Bayes is a Bayesian network with one root node that represents the class and n leaf nodes that represent the attributes. Naïve Bayes classifier is defined as $N(a) = argmax_{c\in C} P(c) \prod_{i=1}^{n} P(x_i|c)$, where $a = \{X_1 = x_1, \ldots, X_n = x_n\}$ is a complete set of attributes In Gaussian Naive Bayes, each attribute is defined by a Gaussian probability density function (PDF) as $X_i \sim N(\mu, \sigma^2)(x) = \frac{1}{\sqrt{2\pi\sigma^2}}$, where μ is the mean and σ^2 is the variance.

Random Forest classifier is a machine learning method that leverages decision trees and ensemble learning. Indeed, the forests are a collection of tree-structured classifiers $\{h(x, \Theta_k),\ k = 1, \ldots\}$, where $\{\Theta_k\}$ independent identically distributed random vectors and each tree are assigned a vote for the most popular class at input x. In fact, the prediction can be made based on majority voting or weighted voting. Additionally, Random Forests can use a large number of attributes and therefore do not require feature selection. Another advantage of this classifier is its resistance to overfitting. However, it heavily depends on the implemented random generator and is deficient in model interpretability.

k-Nearest Neighbors is a popular machine learning method that does not have a learning phase but instead memorizes the training data. Indeed, to predict the class of an unseen instance, the KNN classifier measures the similarity between data points by using the Euclidean distance $d(x,y) = \sqrt{\sum_{k=1}^{n} (x_k - y_k)^2}$, where x_k, y_k are featured elements of instances x and y, respectively.

Knowledge-based models

Graph theory In a cyber security setting, graphs can describe attack prerequisites (vulnerabilities) or attack pathways. The algorithm of finding the shortest path in the tree determines the system exploitability index or optimal route of attack from the attacker's standpoint. In fact, complementing the above graphs with countermeasures aids defense prioritization.

Game theory is the mathematical modeling of interactions among agents. The formal theory defines a game as $Game = (P_i,\ S,\ s,\ \pi_i)$, where P_i stands for players ($i = 1, 2, \ldots$), S is a set of pure strategies for each player I, $s : S_1 \times S_2$ is a set of pure strategy profile, and $\pi_i : S \rightarrow \mathbb{R}$ is the players' payoff functions. Further, the solution to a game is represented as optimal decisions of the players, who may have mutual or conflicting interests.

Markov decision process (MDP) is a stochastic process that is defined as a tuple $(S,\ A,\ P_a,\ R_a)$, where S is a finite set of states, A is a finite state of actions, P_a is the probability that action a in state s at the time t will lead state s' at time $t + 1$ and R_a is a reward expected to be received after transition from state s to s' due to action a. Moreover, the outcome of MDP is a policy π that maps each state to an action a taken in this state s. Additionally, the process has an important property; the action only depends on the current state, not on the prior history. Further, the policy may be realized through a lookup table, or may involve extensive computation [93].

State Machine is an abstract model that represents how the output is computed based on the input. Indeed, the model is formulated mathematically as $SM = (\Sigma,\ S,\ s_0,\ \delta,\ F)$, where Σ a finite set of symbols, S is a finite set of states, s_0 is an initial state of S, δ is a state transition function $\delta : S \times \Sigma \rightarrow S$ and F is a finite set of final states.

Stochastic Activity Networks (SAN) [94] are used for performance, dependability, and performability evaluations. As a stochastic extension of Petri nets, SAN consists of the following elements: places, gates, and activities. Indeed, the gates connect places to activities (input gates) and activities to places (output gates). Additionally, the activities can be instantaneous and timed, which have the delay to completion. Moreover, formally, $SAN = ((P, A, I, O, \gamma, \tau, \iota, o), \mu_0, C, F, G)$, where P is a finite state of places, A is a finite state of activities, I is a finite state of input gates, O is a finite state of output gates, γ is a number of cases for each activity, τ specifies the type of activity, ι maps input gates to activities and o maps the output gates to places.

Stochastic Well-formed Nets (SWN) [95] is a system model that captures the main characteristics of complex systems with the large number of interconnected components. Mathematically, it is defined as $SWN = (WN,\ \theta)$, where WN is a well-formed colored Petri net and θ is a function of transitions.

Competing Risks Theory [96] assesses a specific risk in the complex presence of other k risks and attempts to predict the consequences of removing this risk.

Monte Carlo Simulation is a mathematical method of generating random variables for risk or uncertainty modelling of a certain system.

We map the reviewed methods and their theoretical background in Table 2.

Scope

To establish a connection between the identified threat classes and the methods supporting their visibility, we map them in Table 3. Indeed, the threats related to infrastructure and data received the most attention. We note that resource infiltration, including user credentials, is rarely researched. Similarly, threats such as data tampering and data misuse are also insufficiently studied. In fact, the latter is surprising given the worldwide growth of ransomware attacks against smart cities. Moreover, with rare exceptions, the reviewed works cover one threat category, leaving other

Table 2 Mapping of theoretical background and reviewed methods

Theoretical background	Interdependency models	Risk assessment and threat intelligence	Attack detection
Deep autoencoder (DAE)			[81]
Deep Belief Networks (DBNs)			[77]
Convolutional network			[78]
Fuzzy pattern tree			[79]
Generative Adversarial Networks (GANs)			[85]
Gaussian Naive Bayes			[80]
Random Forest classifier			[80]
k-Nearest Neighbors		[68, 70]	[80, 86]
Graph theory	[48–51, 55, 57]	[58, 60, 62, 63, 67]	
Game theory	[57]		
Markov decision process/chain	[49]	[66]	
State machine	[53, 54, 56]		
Stochastic Activity Networks (SAN)	[52–54]		
Stochastic Well-formed Nets (SWN)	[52]		
Competing Risks Theory	[53]		
Monte Carlo Simulation	[53]		
Various data mining methods		[59, 65, 69, 71, 73, 75, 76]	[82]

Table 3 Methods supporting cyber situational awareness for smart cities

Threat category	Interdependency models	Risk assessment and threat intelligence	Attack detection
Credentials theft	–	[75]	–
Resource infiltration	–	[73, 75]	–
Infrastructure tampering	–	[59, 73, 75]	[80–82]
Resource overwhelming	[52, 53]	[62, 67]	[83–85]
Identity manipulation	–	[62, 69]	[83, 84]
Infection by malware	–	[75]	[78, 79]
Data tampering	–	–	[86]
Data corruption	–	[58, 62, 77]	[76, 77, 83]
Data misuse	–	[63]	–
Disruption of decision-making process	–	–	[87, 88]
Overall threat factor/impact	[49–51, 53–55]	[60, 66–68, 70, 71]	–

threats unaddressed. Further, their limited scope might impede the transition to practical applications.

Visual support

Just as significant, analytics-driven cyber security needs to offer visual support in order to engage human cognition for data interpretation. Visual analytics connect computational data analysis methods and human reasoning in the decision-making process through visualization and interaction. Indeed, the graphical representation conveys a broad spectrum of visual aids to understand how the model works, to represent the results in an intuitive, self-explanatory way, and to enable interaction for visual data

exploration. While the reviewed papers have not been dedicated to producing a visual aid, we compare visual dimensions to understand the role of visualization for analytical analysis. To this end, we derive (and focus) on three categories, namely, performance visualization, model explanation, and knowledge extraction.

(i) *Performance visualization* refers to the graphical representation of the model's accuracy, including the one achieved by employing different parameters in the model.

(ii) *Model explanation* refers to the process of interpreting the discovered knowledge in the form of visual graphics. The first thing to consider here is the visualization of model architecture, in particular, how the model and dataflow are designed. Additionally, computational graphs and flowcharts sufficiently capture the architecture. Moreover, other components to visualize are the model parameters, the contribution of different inputs (i.e., features), and the error measurements (e.g., those generated by adversarial network samples at each step).

(iii) *Knowledge extraction* leverages human cognition which enables users to interpret data and formulate hypotheses more efficiently. In fact, interaction techniques, such as detail-on-demand, dynamic queries, and zooming can significantly improve this process.

Figure 9 illustrates the distribution of the types of the offered visualizations. Even though around 50% of the works still do not leverage a visualization method, researchers found a way to visually clarify a model as a means to explain a method. Indeed, scatterplots, line and bar charts are gradually used as visual structure for model explanation. In fact, the majority of reviewed works used spatial view in the form of 2-dimensional data representation, while the combination of a physical and 2-dimensional structure is only used in one paper. Further, less than 40% of the surveyed models support result visualization. Among them, only one work offers interactivity, while the remaining works solely rely on non-interactive representations.

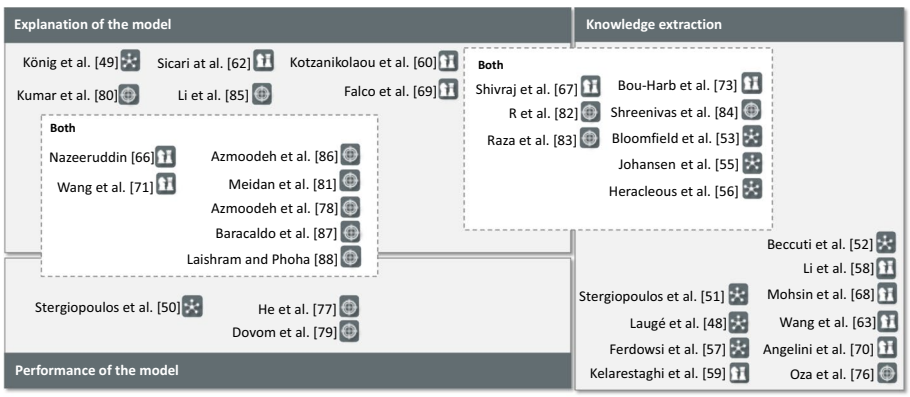

Fig. 9 The distribution of visualization types. The papers that consider two different visualization categories are illustrated close to the line that distinguishes two categories. Labels: —employed for dependency models, —employed for risk assessment, —employed for attack detection

While automated algorithms make pattern recognition, classification, and other functions possible, the combination of these algorithms with visual analytics can undoubtedly enhance the decision-making process.

Underlying data: input, interpretation, and datasets

The reviewed works established a basis for researchers by obtaining underlying data in two ways: (i) using existing datasets and (ii) harvesting data by setting up specific environments in laboratory settings. The former methods are quite efficient given that they avoid any data collection. However, we witness a shortage of smart cities-related datasets. Therefore, the second method of data capture is highly thought-after. Nevertheless, it is typically only suitable for short-term collection, hardly reproducible, and barely covers the entire infrastructure of smart cities.

Table 4 provides a summary of data inputs, the output (interpretation) of the model, and the used dataset for validation.

Given the lack of public datasets that are created for smart cities, the general examples and several datasets are produced in laboratory environments. However, such settings do not capture a smart cities' context. Therefore, they are rarely based on realistic assumptions; thus, their practical implementation might not always be successful or representative.

We now give a brief description of the employed public datasets.

2016 SWaT dataset [98] supports research in the design of secure Cyber-Physical Systems (CPS). Indeed, the data collection was performed on a six-stage Secure Water Treatment (SWaT) testbed that depicts a scaled-down version of an industrial water treatment plant. Additionally, the dataset consists of two behavioral models collected during normal operation and an under attack system. Moreover, the physical properties of the data along with the network traffic contain attacks carried out by the researchers and provides accurate data labels for subsequent use.

Shodan [99] is a search engine for Internet-connected devices. It crawls the Internet 24/7 and updates its repository in real-time to provide a recent list of IoT devices. Additionally, by grabbing and analyzing banners and device meta-data, the engine explores their corresponding various vulnerabilities (including Heartbleed, Logjam, and default passwords).

BullGuard's IoT Scanner is an online search engine that leverages Shodan's service in order to allow users to scan their networks for vulnerabilities.

A *darknet* [100] (also commonly referred to as a network telescope) is a set of routable and allocated yet unused IP addresses. From a design perspective, a darknet is transparent and indistinguishable compared with the rest of the Internet space. From a deployment perspective, it is rendered by network sensors that are implemented and dispersed on numerous strategic points throughout the Internet. The aim of a darknet is to provide a lens on Internet-wide unsolicited traffic; since darknet IP addresses are unused, any traffic targeting them represents anomalous traffic.

All ransomware and malware samples are collected from *VirusTotal*. Indeed, this service combines the output from various antivirus programs and online scan engines to test whether the behavior of the software indicates malicious activities or not.

Table 4 Summary of base data and datasets

Reference	Base data																				Interpretation	Dataset
	System components	Type of components	Physical topology	Functional relations	Stochastic assoc.	List of IoT devices	List of vulnerabilities	Probability of attack	Attack tree	Countermeasures	Real-world attack	Impact model	Sensor readings	Actuator state	Load profile	Power usage	OpCode	RPL DODAG	ETX	Network activity	DNS traffic		
Laugé et al. [48]	✓											✓										Level of interdependencies	Expert opinion
König et al. [49]	✓	✓	✓																			Level of interdependencies	Lab simulation. Expert opinion
Stergiopoulos et al. [50]	✓		✓																			Priority for mitigation	Lab simulation
Stergiopoulos et al. [51]	✓		✓																			Evolution of dependencies	Lab setup
Beccuti et al. [52]				✓	✓																	Level of interdependencies	Testbed presented in [97]
Bloomfield et al. [53]			✓	✓	✓																	Impact of coupled subsystem	Live infrastructure
Netkachov et al. [54]	✓				✓																	Impact on the system under study	A power transmission network
Johansen et al. [55]	✓		✓																			Level of interdependencies	A network of interdependent water, power, and gas systems in Shelby County, Tennessee
Heracleous et al. [56]																							
Ferdowsi et al. [57]	✓		✓																			Priority for mitigation	Generic example
Li et al. [58]							✓		✓	✓												Priority for mitigation	Lab simulation
Kelarestaghi et al. [59]											✓	✓										Impact assessment	Collection of publications
Kotzanikolaou et al. [60]	✓		✓									✓										Priority for mitigation	Lab simulation
Sicari at al. [62]							✓		✓	✓												Exploitability level	General example
Wang et al. [63]							✓		✓	✓												Ranked vulnerabilities	Lab simulation
Radanliev et al. [65]						✓																Economic impact	BullGuard's IoT Scanner
Nazeeruddin [66]	✓	✓					✓	✓	✓													Risk level	General example
Shivraj et al. [67]	✓						✓	✓	✓													Risk level	Lab simulation
Mohsin et al. [68]	✓	✓					✓	✓	✓													Risk level	General example
Falco et al. [69]									✓													Attack plan	General example
Angelini et al. [70]	✓								✓			✓										Impact on the process mission	Live power distribution system
Wang et al. [71]	✓	✓																				Contextualized threat intelligence	General example
Bou-Harb et al. [73]						✓														✓		Contextualized threat intelligence	Darknet; Shodan
Dowling et al. [75]																				✓		Attack vectors and patterns	Honeypot

Additionally, through the public API, the users can automatically upload and verify their files.

Model comparison and evaluation metrics

Interdependency models

To compare the methods introduced in "Interdependency models" section, we consider the following criteria: *categories of dependencies*, *completeness*, and *modeling approach*.

i. *Categories of dependencies* this criterion refers to the types of interdependencies that each method models. In this context, we classify these as cyber, physical, and functional dependencies [101]. Additionally, we label the modeled dependencies as cyber if the state of one domain depends on information transmitted by another. Moreover, physical dependencies represent the networks that share a physical flow. Further, functional dependencies consider, in (or the availability of) one domain, the effect of degradation on the performance of the dependent infrastructure.
ii. *Scale* this criterion measures the coverage of the reviewed methods. Clearly, the number of considered dependent domains directly affects the completeness of the modeled architecture.
iii. *Modeling approach* this criterion labels the method based on the technique employed to model dependencies; probabilistic (P), expert opinion (E), or deterministic (D).

Table 5 summarizes the characteristics above and compares the surveyed methods by producing the following observations. First, all the approaches concentrate on the downstream dependencies and maintain a high-level of detail where each domain is represented as one entity. Indeed, this might indicate that cyber interdependencies require an additional modeling approach in order to analyze the effect of cyber attacks targeting the data layer. Second, the surveyed methods support a limited number of domains which implies that these methods are developed as a proof of concept, and that the feasibility of a practical implementation requires further investigation. In fact, the scalability

Table 5 Comparison of the methods modeling dependencies between smart cities' elements

References	Category			Scale			Modeling approach		
	Cyber	Physical	Functional	< 3	3–5	> 5	P	E	D
Laugé et al. [48]			✓			✓		✓	
König et al. [49]			✓	✓			✓		
Stergiopoulos et al. [50]			✓	✓				✓	
Stergiopoulos et al. [51]			✓			✓		✓	
Beccuti et al. [52]			✓	✓			✓		
Bloomfield et al. [53]		✓	✓	✓			✓		✓
Netkachov et al. [54]	✓			✓			✓		✓
Johansen et al. [55]	✓	✓			✓		✓		
Heracleous et al. [56]					✓				
Ferdowsi et al. [57]	✓	✓		✓			✓		

Modeling approaches: *P* probabilistic, *E* expert opinion, *D* deterministic

of the method and its accuracy should be carefully considered, even though the accuracy evaluation measures remain immature.

Risk assessment methods

One of the most significant challenges of prioritization methods in a smart city's settings is the evaluation of the proposed approach. This issue is confirmed by the absence of standard evaluation metrics in the reviewed papers. Since the selection of risk assessment methodology depends on the system infrastructure, security requirements, and purpose [102], we define herein the following set of metrics to evaluate the sufficiency of each model.

i. *Perspective* this criterion focuses on the resource level that is used to identify the risk. It can be described as three categories; asset-driven, service-driven, and business-driven method [102]. First, the asset-driven category identifies risks associated with smart cities' assets, such as IIoT and IoT devices, cloud services, or software, to name a few. Second, the service-driven models identify the risk in the smart cities' provided services. For instance, the risk can be assessed for smart transportation. Lastly, the business-driven risk assessment concentrates on the business processes.
ii. *Application area* the intertwined architecture of smart cities makes the estimation of an impact way harder than in traditional ICT environments. Indeed, such infrastructure consists of a myriad of heterogeneous devices, communication protocols, and big data ecosystems, not to mention the strong relationship between the elements of the architecture.
iii. *Cybersecurity scope* typically, the cyber security scope refers to the impact on main cyber security objectives, namely, confidentiality, integrity, availability, and accountability. However, to be consistent with the threats described in "Threat landscape", we classify the scope based on four previously identified classes; exploratory threats, infrastructure sabotage, data manipulation, and third-party breaches.
iv. *Threat identification strategy* we observed two main approaches; manual and automatic. The latter, however, relies on third-party databases or platforms.
v. *Uncertainty handling* we extracted two strategies that the reviewed models employ to handle uncertainty. These are probabilistic and ordinal strategies. Indeed, the widely used probabilistic method has well-defined mathematical properties. Additionally, the ordinal measure is represented by ranking the exploitability level of the attack vector. In fact, this ranking is chosen on a scale of 1–9, where 1 is the most difficult path.
vi. *Produced outcome* we observe two main risk calculation methods. Indeed, the framework either computes the exploitability level (not the risk) or employs the classical way that takes into account a likelihood of exploitation and a potential impact indicator. Additionally, the choice of the method that computes risk implies that the models are developed for general-purpose. We note that this method omits other angles of risk evaluation such as financial interpretation, risk assessment for compliance, or safety reasons. Although several publications do not directly state the supported decisions, the produced output implies that the safe-

guard prioritization is a key outcome. In fact, it confirms our previous conclusion that the reviewed methods are general-purpose and omit the prioritization of investments and compliance.

vii. *Credibility* this criterion measures the ability of the approach to capture the real risk level. Indeed, the credibility of the risk assessment model can be measured as reliability and validity. While the former is concerned with the consistency of the results, validity deals with the resulting accuracy compared to the true underlying risk [103].

Table 6 combines the metrics for each selected method and leads to the following remarks. First, the majority of the reviewed methods focus on an asset-based approach, even though the effect on smart cities' operations is still at its infancy. Second, the loss of power due to the exploitation of the power grid can lead to a degradation of the traffic's control's system's performance. At the same time, the reviewed methods do not take into account this dependency. However, the accurate impact estimation seems to be unconceivable due to the lack of empirical data. Besides, all the methods fail to take into account an emerging risk of using infrastructure as an attack platform [104]. Third, the assessment in the reviewed works centers on infrastructure and data manipulation, with a significant bias to the former class. However, the exploratory and third-party threats are the entry points for many attacks while risk assessment methods seem to be undervalued (in terms of their significance). Forth, the reviewed models choose probabilistic and ordinal measures with the nearest frequency, knowing that a game-theoretic approach is rarely explored. Finally, although the reviewed papers seldom measure reliability (in fact, only one model, which is proposed by Falco et al. [69], compared the results with a model produced by experts), all of them omitted the validity measurement.

Attack detection methods

To evaluate the performance of the reviewed models, we measure the following four metrics. The first one refers to the ability of the model to classify the instances correctly, while the second one measures how well the model can capture data patterns. Further, we look at the transparency of the model or how well the process is considered to be trustworthy. A final metric analyzes the capability of the approach to capture the attributes of the detected threats.

The ability of the method to label instances can be presented as *accuracy, precision, recall*, and *F-measure*. Indeed, the estimation of these measures depends on the following indicators.

- True positive (tp) indicates that the positive instance is correctly classified.
- True negative (tn) implies that the negative instance is correctly labeled.
- False positive (fp) indicates that the negative instance is misclassified as positive.
- False negative (fn) indicates that the positive instance is misclassified as negative.
- *Accuracy* is considered to be a prime indicator of the correctness of the detection model. It is calculated as the percentage of all the correctly classified instances to all instances as $(tp + tn)/(tp + tn + fp + fn)$. However, the accuracy can be misleading

Table 6. Evaluation of the prioritization schemes

References	Perspective			Threat enum	Cybersecurity scope				Uncertainty handling	Basis for prioritization
	A	S	B		E	I	D	T		
Li et al. [58]	✓			Manual			✓		⚄	Mitigation prioritization
Kelarestaghi et al. [59]		✓		Manual		✓			N/A	Impact assessment
Kotzanikolaou et al. [60]	✓			Manual	Not specified				⬛	Mitigation prioritization
Sicari et al. [62]	✓			Manual		✓	✓		◠	Exploitability level
Wang et al. [63]	✓			Manual			✓		⬛	Threat factor
Radanliev et al. [65]	✓			Manual		✓				Economic impact
Nazeeruddin [66]		✓		Manual		✓			⬛	Threat factor
Shivraj [67]	✓			Manual		✓			◠	Threat factor
Mohsin et al. [68]	✓			Manual		✓	✓		⬛	Prioritized configuration
Falco et al. [69]	✓			Automatic		✓			◠	Attack path projection
Angelini et al. [70]			✓	Manual	Not specified				◠	Operational impact
Wang et al. [71]			✓	Manual		✓	✓		⬛	Threat factor
Bou-Harb et al. [73]	✓			Automatic		✓			⬛	Situational awareness

Perspective: *A* asset-driven, *S* service-driven, *B* business-driven. Cybersecurity scope: *E* exploratory threat, *I* infrastructure sabotage, *D* data manipulation, *T* third-party breaches. Uncertainty: ◠—ordinal, ⬛—probabilistic, ⚄—game-theoretic

in case of high class imbalance [105]. In this case, the following metrics are required to evaluate a model.

- *Precision* measures the proportion of correctly classified instances of all the records that are classified as positive. It is defined as $tp/(tp+fp)$. Indeed, a low precision can indicate a large number of fp.
- *Recall*, also known as sensitivity or true positive rate, represents the ratio of correctly classified positive instances to a number of instances that should be classified as positive. It is formally defined as $tp/(tp+fn)$. In fact, a low recall indicates a large number of fn.
- *F-Measure* is the harmonic mean of precision and recall. It is defined as $2*tp/(2*tp+fp+fn)$.

The problem with the ability of the model to classify instances correctly is that it does not validate the model's performance on previously unseen data. To this end, we evaluate how well the model captures the data pattern. In fact, several methods generalize a model's performance and help evaluate a model's ability to capture data patterns.

(i) *Hold-out technique* randomly divides the dataset into two subsets, namely, training and testing. The split is usually 60/40, 70/30, or 80/20. To avoid a situation when the uneven distribution of classes is found in a subset, it is essential to balance the instances belonging to the different classes.

(ii) *k-fold cross validation* divides datasets into k subsets; one of them is used as the testing set and the other $k-1$ subsets form the training set. Indeed, the method ensures that each instance is a part of a testing set exactly one time. Additionally, the process of training and testing the model is repeated k times and the average error through all tests is used for evaluation. However, the k-fold cross validation is computationally expensive, because the training and testing process should be repeated k times.

(iii) *Leave-one-out cross validation* is a k-fold cross validation with k equal to the number of data instances in the dataset. The evaluation produced by this method is considered to be good even though it is not optimal in terms of computation.

(iv) *Matthews correlation coefficient (MCC)* measure considers all metrics from the confusion matrix to diminish the influence of one class.

$MCC = (tp*tn-fp*fn)\Big/\sqrt{(tp+fp)(tp+fn)(tn+fp)(tn+fn)}$. Indeed, MCC equal 1 indicates the perfect prediction, while -1 refers to the worst prediction.

Further, machine learning models are often criticized by the users as being black-box due to the lack of interpretability that helps us understand how the models make decisions based on the data. To this end, we evaluate each model's transparency by looking at how the model quantifies the influence of each input, details the model's errors, and records the results at each step of the model. Indeed, with different levels of details, several works visually explained the steps of the proposed methods. However, nearly 50% of the methods are still obscure. In fact, the clarity of the model can boost trust and practical adoption. This being said, more explanation should be given to the interpretability of the results and the process itself.

Table 7 summarizes the metrics that are used to evaluate the reviewed detection methods. Although we report the value of various metrics for each work, they are hardly comparable due the underlying nature of the dataset used for evaluation and the different level of detail that is provided in the reviewed papers.

The numerous models achieved accuracy over 95%. Additionally, the validation methods and used datasets demonstrated a profound effect on accuracy. For instance, the majority of techniques that used artificial datasets or simulated environments reported lower accuracy than their peers that leveraged live data. In fact, the tenfold cross-validation exhibited higher accuracy.

We now consider methods that are capable of capturing threat traits. In this context, we take a look at the explicit outcomes of the detection methods and measure how these outcomes answer the following questions.

- *Detection goal* what kind of attack does the method attempt to detect?
- *Attack phase* at what phase of attack the method detects an intrusion?
- *Attack vector* does the method analyze how the attack was facilitated?
- *Attribution* does the method attribute the attack to a specific adversary?
- *Time-to-detection (TTD)* how long does it take to detect an attack?
- *Impact* does the method analyze potential attack impact?

Table 8 summarizes the details that can be extracted from the analyzed methods.

Open questions and future perspective

Although we already mentioned some open research questions, in this section, we encapsulate them and elaborate on several possible research directions that can address these issues.

Further, in the context of smart cities, cyber threats and attacks, which are induced by exploiting heterogeneous advanced technologies, are indeed evolving rapidly. Thus, failing to manage these cyber threats impairs the trustworthiness of smart cities' endeavors.

Table 7 The performance benchmarks of various threat detection methods

References	Accuracy (%)	Precision (%)	Recall (%)	F-Measure	TP	FP	Validation
Oza et al. [76]	–	–	–	–	–	–	–
He et al. [77]	93.73–98.51	–	–	–	–	–	–
Azmoodeh et al. [78]	99.68	98.59	98.37	0.98	–	–	Tenfold
Dovom et al. [79]	96.4	94.33	89.71	0.89	–	–	MCC
Kumar et al. [80]	94.44	92	1	0.96	–	–	Hold-out
Meidan et al. [81]	–	–	–	–	100	0.7	Hold-out
R et al. [82]	99	85	99	92			Hold-out
Raza et al. [83]	80–100	-	100	-			-
Shreenivas et al. [84]	90–100	–	–	–			-
Li et al. [85]	94.8	93	63.64	0.75			Hold-out
Azmoodeh et al. [86]	94.27	89.19	95.65	0.92			Leave-one-out
Baracaldo et al. [87]	Up to 90	–	–	–			Hold-out
Laishram and Phoha [88]	Up to 99	–	–	–			Tenfold

Table 8 Attack details provided by the detection method

Reference	Detection goal	Attack phase	TTD
Oza et al. [76]	False data injection	Action	
He et al. [77]	False data injection	Installation	
Azmoodeh et al. [78]	Malware	Exploitation	
Dovom et al. [79]	Malware	Exploitation	
Kumar et al. [80]	Probing detection	Reconnaissance	
Meidan et al. [81]	Botnet	Action	174 ± 212 ms
R et al. [82]	Botnet	Action	
Raza et al. [83]	Routing attacks False data injection	Action	
Shreenivas et al. [84]	Routing attacks	Action	
Li et al. [85]	Anomaly	Various	
Azmoodeh et al. [86]	Ransomware (IoT)	Exploitation	
Baracaldo et al. [87]	Poisoning attack	Exploitation	
Laishram and Phoha [88]	Poisoning attack	Exploitation	

Therefore, it is imperative to acknowledge a proactive approach in order to secure different levels of a smart city's architecture. Additionally, since there's a shortage in the security-related budget, methods should prioritize spending in order to boost the resilience of the entire ecosystem. Although several methods support this imperative task, there are a number of observations (O) that require attention from the research community.

O1 *The lack of holistic framework for situational awareness.* Cyber situational awareness is indeed a challenging task. In fact, the reviewed methods contribute to a single component of smart cities' architecture without modeling the dependencies among them. Additionally, there seem to be no holistic solution to address the prioritization threat in the context of specific infrastructure (e.g., energy, transportation, health, etc.). Therefore, it is critical to frame identified threats and detected ongoing attacks in the context of smart cities' operation and comprehend their real impact on mission-critical services. However, the relationships are not always straightforward. Thus, the development of a complete solution requires interdisciplinary research.

O2 *Support of threat escalation analysis is challenging.* A threat escalation should be thoroughly investigated to support cyber decisions. Indeed, there are several solutions that could help with prioritizing decisions. First, blending the information regarding the time required for the investigation and the remediation of detected malicious events. Second, the effectiveness of previously applied defense mechanisms to a similar problem. Third, the cost–benefit analysis of mitigation. In this context, more research can be pursued to support the decision-making process for smart cities' security.

O3 *Limited visual analytics for situational awareness.* One of the biggest challenges of situational awareness is the amount and quality of information that should be analyzed. Although automated methods rooted in machine learning and computational power of modern computers enable effective data processing, the analysis still requires human judgment in order to make the best possible evaluation of the result and eliminate the negative effect of conflicting or incomplete data. Indeed, visual analytics connect computational data analysis methods and human reasoning in the decision-making process through visualization and interaction.

Such integration, known as visual analytics, is largely perceived by the research community [106]. In fact, it synthesizes information to derive insights and communicate the assessment for prompt response. However, the usage of human cognition to identify and track threats' progress, evaluate supporting information, and enhance decision-making seems to be in its infancy in the context of smart cities'. Surprisingly, limited amount of reviewed works made an attempt to visualize the results, even though it can allow cyber analysts to accomplish their responsibilities with a more comprehensive support. Without such capabilities, the practical implementation of the analytical models is problematic, especially since smart cities have such a complex environment.

O4 *Evaluation of threat prioritization models is challenging*. One of the most significant challenges of threat exploration methods in smart cities' settings is their evaluation. Indeed, limited visibility of dependencies between elements in the entire ecosystem, continually evolving threats, and access to past cyber security incidents make it challenging to establish a ground truth. Additionally, most of the reviewed methods validated the results through generic illustrative scenarios. However, the lack of connection with real-world applications questions the validity of the evaluated approaches. Moreover, the reliability of the proposed methods is also rarely measured because of the broad lack of empirical data (for comparisons). Therefore, the application of field strategies such as interviews, experiments, and similar studies can be instrumental in addressing the evaluation task.

O5 *Data gap*. Despite advances in the field of cyber security, the main challenge of generalizing knowledge derived from the limited collection of previously inferred malicious events related to smart cities remain unsolved. Indeed, the evaluation datasets play a vital role in validating the approaches. Due to the lack of publicly available raw data regarding events and their impact on various aspects of smart cities, the models are evaluated based on the data generated in laboratory setups. Additionally, it appears that the reviewed methods enumerate threats and attacks manually, without formal representation, not to mention the absence of sharing capabilities. Moreover, generating, maintaining, and sharing the knowledge-base regarding attack plans can be a possible solution to this issue. Further, with the increasing number of malicious incidents, the systematic approach of collecting, indexing, and correlating incidents enables comprehensive situational awareness, faster detection, and mitigation. Therefore, establishing relevant datasets with a sufficient amount of data, broad scope, and an even number of attack types can support the solution of the evaluation problem and improve the threat scope. Additionally, while considering the ethical aspect, sharing raw data is a candidate for a possible solution to this issue.

Conclusion

In this article, we presented a literature survey of methods that support the visibility of cyber threats in the context of smart cities. We first synthesized the threats against smart cities, linked them to attack types, and discussed their potential impacts. We then described and evaluated the methods dedicated to modeling dependencies among various infrastructure of smart cities, risk assessment methods, and attack detection techniques. We also compared and contrasted the methods in each category, discussed findings and related issues, learned lessons, and suggested possible

future research directions. Three important findings emerged from the literature review.

First, cyber situational awareness in the context of smart cities seems to be in a juvenile stage. Indeed, cyber dependencies between the various components of smart cities' infrastructure are not thoroughly studied. However, identified threats and ongoing attacks should be put in the context of smart cities' operations, and should consider interdependencies among domains in order to realize a real impact on mission-critical services. Additionally, more interdisciplinary research is required to capture the dependencies, including those that are cyber-related, between different components of the smart cities' ecosystem. Moreover, cyber dependencies, safety, financial, and operational effects should be investigated to realize the impact of cyber-attacks on different components. Further, from the visual perspective, effective representation might be needed to capture such cyber dependencies.

Second, cyber-related data for smart cities is increasingly unavailable. In fact, establishing relevant datasets with a broad scope and sufficient amount of raw data can provide a solution for the evaluation problem and improve threat landscape's visibility.

Third, in the context of smart cities, more attention should be given to the evaluation of the reviewed methods' credibility and transparency. Indeed, by doing so, we could transition from these methods to a more practical implementation. However, the reliability metrics of threat prioritization techniques are not well established yet. Additionally, it is practically impossible to establish a ground truth due to many reasons. First, the visibility of interdomain dependencies in the entire ecosystem is limited. Second, accessing past cyber security incidents is hard. Third, empirical data for comparison is minimal. For the visual analytics community, it could symbolize the creation of visual techniques to reveal the insights of machine learning models or to create a visual representation of threats progression through the entire system of smart cities.

Indeed, methods supporting cyber situational awareness attract researchers in many ways; from modeling dependencies to assessing risks and detecting attacks. Although it appears that ongoing research lacks empirical data to establish sound ground truth, it is indeed a critical area that requires interdisciplinary exploration, perhaps with the help of industry bodies.

Acknowledgements
The authors would like to thank the anonymous reviewers for their constructive evaluation of this paper.

Authors' contributions
NN performed the primary literature review and analysis for this work, and also drafted the manuscript. CN reviewed and edited the article. EBH worked with NN to develop the article's framework and focus. BF introduced this topic to NN and worked with her to develop article's framework and focus. All authors read and approved the final manuscript.

Funding
This work was supported by the Florida Center for Cybersecurity (Cyber Florida); U.S. National Science Foundation (NSF) [Office of Advanced Cyberinfrastructure (OAC)].

Availability of data and materials
Not applicable.

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Author details
[1] Department of Computer and Electrical Engineering and Computer Sciences, Florida Atlantic University, Boca Raton, USA. [2] The Cyber Center For Security and Analytics, University of Texas at San Antonio, San Antonio, USA.

Received: 30 July 2020 Accepted: 8 October 2020
Published online: 21 October 2020

References

1. United Nations. 68% of the world population projected to live in urban areas by 2050. 2018. https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html. Accessed 15 Apr 2020.
2. City Profile. Smart cities world. https://www.smartcitiesworld.net/smart-cities-profile. Accessed 25 Apr 2020.
3. Singapore uses IoT to create smart buildings. 2016. https://www.smart-energy.com/regional-news/asia/singapore-iot-smart-buildings/. Accessed 15 Apr 2020.
4. Building a smart + equitable city. The official website of the City of New York. 2015. https://www1.nyc.gov/assets/forward/documents/NYC-Smart-Equitable-City-Final.pdf. Accessed 05 Apr 2020.
5. IBM. City of Rio de Janeiro and IBM collaborate to advance emergency response system; access to real-time information empowers citizens. 2011. https://www.prnewswire.com/news-releases/city-of-rio-de-janeiro-and-ibm-collaborate-to-advance-emergency-response-system-access-to-real-time-information-empowers-citizens-133545433.html. Accessed Apr 09 2020.
6. Mclaughlin T. As shootings soar, Chicago police use technology to predict crime. 2017. https://www.reuters.com/article/us-chicago-police-technology/as-shootings-soar-chicago-police-use-technology-to-predict-crime-idUSKBN1AL08P. Accessed 08 Apr 2020.
7. The Register. Sweden 'secretly blames' hackers—not solar flares—for taking out air traffic control. The Register. 2018. https://www.theregister.co.uk/2016/04/12/sweden_suspects_russian_hackers_hit_air_traffic_control/. Accessed 03 Mar 2020.
8. Case DU. Analysis of the cyber attack on the Ukrainian power grid. Electricity Information Sharing and Analysis Center (E-ISAC), vol. 388, 2016.
9. Kraszewski K. SamSam and the Silent Battle of Atlanta. In: 2019 11th international conference on cyber conflict (CyCon), 2019. vol. 900, p. 1–16.
10. Kan M. Ransomware strikes Baltimore's 911 dispatch system. PCMag Asia. 2018. https://sea.pcmag.com/news/20374/ransomware-strikes-baltimores-911-dispatch-system. Accessed 04 Apr 2020.
11. Mettler K. Somebody keeps hacking these Dallas road signs with messages about Donald Trump Bernie Sanders and Harambe the gorilla. Washington, DC: WP Company; 2019.
12. Dallas warning sirens "set off by hacker". BBC. 2017.
13. Khan R, Kumar P, Jayakody DNK, Liyanage M. A survey on security and privacy of 5G technologies: potential solutions, recent advancements and future directions. IEEE Commun Surv Tutor. 2019;22(1):196–248.
14. Chan L, et al. Survey of AI in cybersecurity for information technology management. In: 2019 IEEE technology & engineering management conference (TEMSCON). 2019. p. 1–8.
15. Druzdzel MJ, Flynn RR. Decision support systems. In: Encyclopedia of library and information sciences. Boca Raton: CRC Press; 2017. p. 1200–8.
16. Ijaz S, Shah MA, Khan A, Ahmed M. Smart cities: a survey on security concerns. Int J Adv Comput Sci Appl. 2016;7(2):612–25.
17. Gharaibeh A, et al. Smart cities: a survey on data management, security, and enabling technologies. IEEE Commun Surv Tutor. 2017;19(4):2456–501. https://doi.org/10.1109/COMST.2017.2736886.
18. Silva BN, Khan M, Han K. Towards sustainable smart cities: a review of trends, architectures, components, and open challenges in smart cities. Sustain Cities Soc. 2018;38:697–713.
19. Baig ZA, et al. Future challenges for smart cities: cyber-security and digital forensics. Digit Investig. 2017;22:3–13.
20. Cui L, Xie G, Qu Y, Gao L, Yang Y. Security and privacy in smart cities: challenges and opportunities. IEEE Access. 2018;6:46134–45.
21. Sookhak M, Tang H, He Y, Yu FR. Security and privacy of smart cities: a survey, research issues and challenges. IEEE Commun Surv Tutor. 2019;21(2):1718–43. https://doi.org/10.1109/COMST.2018.2867288.
22. Talari S, Shafie-Khah M, Siano P, Loia V, Tommasetti A, Catalão JP. A review of smart cities based on the internet of things concept. Energies. 2017;10(4):421.
23. Banerjee J, Das A, Sen A. A survey of interdependency models for critical infrastructure networks. ArXiv Prepr. ArXiv170205407. 2017.
24. Tøndel IA, Foros J, Kilskar SS, Hokstad P, Jaatun MG. Interdependencies and reliability in the combined ICT and power system: an overview of current research. Appl Comput Inform. 2018;14(1):17–27.
25. Kitchin R, Dodge M. The (in) security of smart cities: vulnerabilities, risks, mitigation, and prevention. J Urban Technol. 2019;26(2):47–65.
26. Vitunskaite M, He Y, Brandstetter T, Janicke H. Smart cities and cyber security: are we there yet? A comparative study on the role of standards, third party risk management and security ownership. Comput Secur. 2019;83:313–31.
27. Habibzadeh H, Nussbaum BH, Anjomshoa F, Kantarci B, Soyata T. A survey on cybersecurity, data privacy, and policy issues in cyber-physical system deployments in smart cities. Sustain Cities Soc. 2019;50:101660.
28. Mehmood Y, Ahmad F, Yaqoob I, Adnane A, Imran M, Guizani S. Internet-of-things-based smart cities: recent advances and challenges. IEEE Commun Mag. 2017;55(9):16–24. https://doi.org/10.1109/MCOM.2017.1600514.

29. Galluscio M, et al. A first empirical look on internet-scale exploitations of IoT devices. In: 2017 IEEE 28th annual international symposium on personal, indoor, and mobile radio communications (PIMRC). 2017. p. 1–7.
30. Ercolani VJ, Patton MW, Chen H. Shodan visualized. In: 2016 IEEE conference on intelligence and security informatics (ISI). 2016. p. 193–5.
31. Patton M, Gross E, Chinn R, Forbis S, Walker L, Chen H. Uninvited connections: a study of vulnerable devices on the internet of things (IoT). In: 2014 IEEE joint intelligence and security informatics conference. 2014. p. 232–5.
32. PALO ALTO NETWORKS. Impacts of cyberattacks on IoT devices. https://www.sdxcentral.com/wp-content/uploads/2019/10/iot-research-paper-v4_final.pdf. Accessed 04 Apr 2020.
33. Sicato S, Costa J, Sharma PK, Loia V, Park JH. VPNFilter malware analysis on cyber threat in smart home network. Appl Sci. 2019;9(13):2763.
34. Zimba A, Wang Z, Mulenga M. Cryptojacking injection: a paradigm shift to cryptocurrency-based web-centric internet attacks. J Organ Comput Electron Commer. 2019;29(1):40–59.
35. Bou-Harb E, Debbabi M, Assi C. A novel cyber security capability: inferring internet-scale infections by correlating malware and probing activities. Comput Netw. 2016;94:327–43.
36. Bertino E, Islam N. Botnets and internet of things security. Computer. 2017;50(2):76–9.
37. Kumar M. DDoS attack takes down central heating system amidst winter in Finland. The Hacker News. 2016. https://thehackernews.com/2016/11/heating-system-hacked.html. Accessed 04 Apr 2020.
38. Trappe W, Howard R, Moore RS. Low-energy security: limits and opportunities in the internet of things. IEEE Secur Priv. 2015;13(1):14–21.
39. Georgiou K, Xavier-de-Souza S, Eder K. The IoT energy challenge: a software perspective. IEEE Embed Syst Lett. 2017;10(3):53–6.
40. Mohurle S, Patil M. A brief study of wannacry threat: ransomware attack 2017. Int J Adv Res Comput Sci. 2017. https://doi.org/10.26483/IJARCS.V8I5.4021.
41. Ransomware attack on San Francisco public transit gives everyone a free ride. The Guardian. 2016. https://www.theguardian.com/technology/2016/nov/28/passengers-free-ride-san-francisco-muni-ransomeware. Accessed 25 Apr 2020.
42. Liu Y, Ning P, Reiter MK. False data injection attacks against state estimation in electric power grids. ACM Trans Inf Syst Secur TISSEC. 2011;14(1):1–33.
43. Liang G, Zhao J, Luo F, Weller SR, Dong ZY. A review of false data injection attacks against modern power systems. IEEE Trans Smart Grid. 2016;8(4):1630–8.
44. Wurm J, Hoang K, Arias O, Sadeghi AR, Jin Y. Security analysis on consumer and industrial IoT devices. In: 2016 21st Asia and South Pacific design automation conference (ASP-DAC). 2016. p. 519–24.
45. Van Zoonen L. Privacy concerns in smart cities. Gov Inf Q. 2016;33(3):472–80.
46. Usama M, Asim M, Latif S, Qadir J, et al. Generative adversarial networks for launching and thwarting adversarial attacks on network intrusion detection systems. In: 2019 15th international wireless communications & mobile computing conference (IWCMC). 2019. p. 78–83.
47. Lin P, Swimmer M, Urano A, Hilt S, ve Vosseler R (2017) Securing smart cities moving toward utopia with security in mind. A TrendLabs Research Paper, Erişim Tarihi: 15 Eylül 2019 [Online]. https://documents.trendmicro.com/assets/wp/wp-securing-smart-cities.pdf. Accessed 15 June 2020.
48. Laugé A, Hernantes J, Sarriegi JM. Critical infrastructure dependencies: a holistic, dynamic and quantitative approach. Int J Crit Infrastruct Prot. 2015;8:16–23.
49. König S, Rass S. Investigating stochastic dependencies between critical infrastructures. Int J Adv Syst Meas. 2018;11:250–8.
50. Stergiopoulos G, Kotzanikolaou P, Theocharidou M, Gritzalis D. Risk mitigation strategies for critical infrastructures based on graph centrality analysis. Int J Crit Infrastruct Prot. 2015;10:34–44.
51. Stergiopoulos G, Kotzanikolaou P, Theocharidou M, Lykou G, Gritzalis D. Time-based critical infrastructure dependency analysis for large-scale and cross-sectoral failures. Int J Crit Infrastruct Prot. 2016;12:46–60.
52. Beccuti M, Chiaradonna S, Di Giandomenico F, Donatelli S, Dondossola G, Franceschinis G. Quantification of dependencies between electrical and information infrastructures. Int J Crit Infrastruct Prot. 2012;5(1):14–27.
53. Bloomfield RE, Popov P, Salako K, Stankovic V, Wright D. Preliminary interdependency analysis: an approach to support critical-infrastructure risk-assessment. Reliab Eng Syst Saf. 2017;167:198–217.
54. Netkachov O, Popov P, Salako K. Quantification of the impact of cyber attack in critical infrastructures. In: International conference on computer safety, reliability, and security. 2014. p. 316–27.
55. Johansen C, Tien I. Probabilistic multi-scale modeling of interdependencies between critical infrastructure systems for resilience. Sustain Resilient Infrastruct. 2018;3(1):1–15.
56. Heracleous C, Kolios P, Panayiotou CG, Ellinas G, Polycarpou MM. Hybrid systems modeling for critical infrastructures interdependency analysis. Reliab Eng Syst Saf. 2017;165:89–101.
57. Ferdowsi A, Saad W, Maham B, Mandayam NB. A Colonel Blotto game for interdependence-aware cyber-physical systems security in smart cities. In: Proceedings of the 2nd international workshop on science of smart city operations and platforms engineering. 2017. p. 7–12.
58. Li Z, Jin D, Hannon C, Shahidehpour M, Wang J. Assessing and mitigating cybersecurity risks of traffic light systems in smart cities. IET Cyber Phys Syst Theory Appl. 2016;1(1):60–9.
59. Kelarestaghi KB, Foruhandeh M, Heaslip K, Gerdes R. Intelligent transportation system security: impact-oriented risk assessment of in-vehicle networks. IEEE Intell Transp Syst Mag. 2019. https://doi.org/10.1109/MITS.2018.2889714.
60. Kotzanikolaou P, Theoharidou M, Gritzalis D. Assessing n-order dependencies between critical infrastructures. Int J Crit Infrastruct. 2013;9(1–2):93–110.
61. Neshenko N, Bou-Harb E, Crichigno J, Kaddoum G, Ghani N. Demystifying IoT security: an exhaustive survey on IoT vulnerabilities and a first empirical look on internet-scale IoT exploitations. IEEE Commun Surv Tutor. 2019;21(3):2702–33.

62. Sicari S, Rizzardi A, Miorandi D, Coen-Porisini A. A risk assessment methodology for the internet of things. Comput Commun. 2018;129:67–79.
63. Wang H, Chen Z, Zhao J, Di X, Liu D. A vulnerability assessment method in industrial internet of things based on attack graph and maximum flow. IEEE Access. 2018;6:8599–609.
64. Mell P, Scarfone K, Romanosky S. Common vulnerability scoring system. IEEE Secur Priv. 2006;4(6):85–9.
65. Radanliev P, et al. Future developments in cyber risk assessment for the internet of things. Comput Ind. 2018;102:14–22.
66. Mohammad N. A multi-tiered defense model for the security analysis of critical facilities in smart cities. IEEE Access. 2019;7:152585–98.
67. Shivraj V, Rajan M, Balamuralidhar P. A graph theory based generic risk assessment framework for internet of things (IoT). In: 2017 IEEE international conference on advanced networks and telecommunications systems (ANTS). 2017. p. 1–6.
68. Mohsin M, Sardar MU, Hasan O, Anwar Z. IoTRiskAnalyzer: a probabilistic model checking based framework for formal risk analytics of the internet of things. IEEE Access. 2017;5:5494–505.
69. Falco G, Viswanathan A, Caldera C, Shrobe H. A master attack methodology for an AI-based automated attack planner for smart cities. IEEE Access. 2018;6:48360–73.
70. Angelini M, Santucci G. Visual cyber situational awareness for critical infrastructures. In: Proceedings of the 8th international symposium on visual information communication and interaction. 2015. p. 83–92.
71. Wang P, Ali A, Kelly W. Data security and threat modeling for smart city infrastructure. In: 2015 international conference on cyber security of smart cities, industrial control system and communications (SSIC), 2015. p. 1–6. https://doi.org/10.1109/SSIC.2015.7245322.
72. Wang SP, Ledley RS. Computer architecture and security: fundamentals of designing secure computer systems. New York: Wiley; 2012.
73. Bou-Harb E, Neshenko N. Cyber threat intelligence for the internet of things. New York: Springer; 2020.
74. Naik DR, Das LB, Bindiya TS. Wireless sensor networks with Zigbee and WiFi for environment monitoring, traffic management and vehicle monitoring in smart cities. In: 2018 IEEE 3rd international conference on computing, communication and security (ICCCS). 2018. p. 46–50.
75. Dowling S, Schukat M, Melvin H. A ZigBee honeypot to assess IoT cyberattack behavior. In: 2017 28th Irish signals and systems conference (ISSC). 2017. p. 1–6.
76. Oza P, Foruhandeh M, Gerdes R, Chantem T. Secure traffic lights: replay attack detection for model-based smart traffic controllers. In: Proceedings of the second ACM workshop on automotive and aerial vehicle security. 2020. p. 5–10.
77. He Y, Mendis GJ, Wei J. Real-time detection of false data injection attacks in smart grid: a deep learning-based intelligent mechanism. IEEE Trans Smart Grid. 2017;8(5):2505–16. https://doi.org/10.1109/TSG.2017.2703842.
78. Azmoodeh A, Dehghantanha A, Choo K-KR. Robust malware detection for internet of (battlefield) things devices using deep Eigenspace learning. IEEE Trans Sustain Comput. 2018;4(1):88–95.
79. Dovom EM, Azmoodeh A, Dehghantanha A, Newton DE, Parizi RM, Karimipour H. Fuzzy pattern tree for edge malware detection and categorization in IoT. J Syst Archit. 2019;97:1–7.
80. Kumar A, Lim TJ. EDIMA: early detection of IoT malware network activity using machine learning techniques. In: 2019 IEEE 5th world forum on internet of things (WF-IoT). 2019. p. 289–94. https://doi.org/10.1109/WF-IoT.2019.8767194.
81. Meidan Y, et al. N-baiot—network-based detection of IoT botnet attacks using deep autoencoders. IEEE Pervasive Comput. 2018;17(3):12–22.
82. Alazab VRM, Srinivasan S, Pham Q, Padannayil SK, Simran K. A visualized botnet detection system based deep learning for the internet of things networks of smart cities. IEEE Trans Ind Appl. 2020. https://doi.org/10.1109/TIA.2020.2971952.
83. Raza S, Wallgren L, Voigt T. SVELTE: real-time intrusion detection in the internet of things. Ad Hoc Netw. 2013;11(8):2661–74.
84. Shreenivas D, Raza S, Voigt T. Intrusion detection in the RPL-connected 6LoWPAN networks. In: Proceedings of the 3rd ACM international workshop on IoT privacy, trust, and security. 2017. p. 31–8.
85. Li D, Chen D, Goh J, Ng S. Anomaly detection with generative adversarial networks for multivariate time series. ArXiv Prepr. ArXiv180904758. 2018.
86. Azmoodeh A, Dehghantanha A, Conti M, Choo K-KR. Detecting crypto-ransomware in IoT networks based on energy consumption footprint. J Ambient Intell Humaniz Comput. 2018;9(4):1141–52.
87. Baracaldo N, Chen B, Ludwig H, Safavi A, Zhang R. Detecting poisoning attacks on machine learning in IoT environments. In: 2018 IEEE international congress on internet of things (ICIOT). 2018. p. 57–64.
88. Laishram R, Phoha VV. Curie: a method for protecting SVM classifier from poisoning attack. ArXiv Prepr. ArXiv160601584. 2016.
89. Goodfellow I, Bengio Y, Courville A. Deep learning. Cambridge: MIT Press; 2016.
90. Hinton GE. Deep belief nets. 2010.
91. Senge R, Hüllermeier E. Fast fuzzy pattern tree learning for classification. IEEE Trans Fuzzy Syst. 2015;23(6):2024–33.
92. Goodfellow I, et al. Generative adversarial nets. In: Advances in neural information processing systems. Cambridge: MIT Press; 2014. p. 2672–80.
93. Edelkamp S, Schrödl S. Chapter 1—Introduction. In: Edelkamp S, Schrödl S, editors. Heuristic search. San Francisco: Morgan Kaufmann; 2012. p. 3–46.
94. Sanders WH, Meyer JF. Stochastic activity networks: formal definitions and concepts. In: School organized by the European Educational Forum. 2000. p. 315–43.
95. Chiola G, Dutheillet C, Franceschinis G, Haddad S. Stochastic well-formed colored nets and symmetric modeling applications. IEEE Trans Comput. 1993;42(11):1343–60.
96. David HA, Moeschberger ML. The theory of competing risks. London: Charles Griffin and Company; 1978.

97. Dondossola G, Garrone G, Szanto J, Deconinck G, Loix T, Beitollahi H. ICT resilience of power control systems: experimental results from the CRUTIAL testbeds. In: 2009 IEEE/IFIP international conference on dependable systems & networks. 2009. p. 554–9.
98. Goh J, Adepu S, Junejo KN, Mathur A. A dataset to support research in the design of secure water treatment systems. In: International conference on critical information infrastructures security. 2016. p. 88–99.
99 Shodan® [Online]. http://shodan.io. Accessed 5 Mar 2020.
100. UCSD network telescope—near-real-time network telescope dataset. https://www.caida.org/data/passive/telescope-near-real-time_dataset.xml. Accessed 05 Mar 2020.
101. Rinaldi SM, Peerenboom JP, Kelly TK. Identifying, understanding, and analyzing critical infrastructure interdependencies. IEEE Control Syst Mag. 2001;21(6):11–25.
102. Shameli-Sendi A, Aghababaei-Barzegar R, Cheriet M. Taxonomy of information security risk assessment (ISRA). Comput Secur. 2016;57:14–30.
103. Aven T, Heide B. Reliability and validity of risk analysis. Reliab Eng Syst Saf. 2009;94(11):1862–8.
104. Nurse JRC, Creese S, Roure DD. Security risk assessment in internet of things systems. IT Prof. 2017;19(5):20–6. https://doi.org/10.1109/MITP.2017.3680959.
105. Xin Y, et al. Machine learning and deep learning methods for cybersecurity. IEEE Access. 2018;6:35365–81.
106. Thomas JJ, Cook KA. A visual analytics agenda. IEEE Comput Graph Appl. 2006;26(1):10–3.

Publisher's Note