Towards scalable intrusion detection

Siraj A. Shaikh, research fellow, Department of Informatics and Sensors, Cranfield University, UK
Howard Chivers, professor, Department of Informatics and Sensors, Cranfield University, UK
Philip Nobles, lecturer, Department of Informatics and Sensors, Cranfield University, UK
John A. Clark, professor, Department of Computer Science, University of York, UK
Hao Chen, research associate, Department of Computer Science, University of York, UK

We raise some questions that need to be addressed to overcome the scalability challenges that intrusion detection systems face. We also take this opportunity to highlight some of the recent related work and put it in context.

While various efforts contribute to addressing some of the individual challenges, they do not promise a coherent way forward. There is a need to consider the problem from a fresh perspective. We propose some broader design principles and strategies that essentially serve as research problems and need to be addressed if system-smart and viable scalable intrusion detection is to be realised.

Increasingly intrusion detection is being required to be scalable. This is necessitated by the huge volumes of traffic monitored at larger and faster modern networks of today, along with the continually growing attack activity on public networks, such as worms, large scale botnets and probing, with tens of thousands of probes increasingly common.1, 7, 17, 23

"Scalability has received relatively little attention in intrusion detection research so far, and can be broadly categorised in terms of network, temporal and traffic scalability."

Rising proportions of non-productive traffic, mostly backscatter, spam or malicious traffic, which is increasingly persistent and originates from a variety of sources including intruders, worm-infected machines, flood response and even misconfigured hosts. 17

Scalability has received relatively little attention in intrusion detection research so far, and can be broadly categorised in terms of network, temporal and traffic scalability.

Network scalability addresses the problems of monitoring large number of nodes. As the number of nodes grows, so does the amount of overall monitoring and the cost of deploying additional sensors.

This means the monitoring infrastructure is often overloaded and operators are buried under data.

Gates et al present an approach to detecting scans for large networks where the characteristics of the network are not entirely known and only unidirectional flow level data is available.9 Antonatos et al consider deploying lightweight honeypots deployed on an adhoc basis by a community of network users.2 This approach to dynamic deployment of a lightweight mechanism promises low cost honeypot monitoring on a large number of nodes without expert configuration.

Temporal scalability aims at the capability to detect slow stealthy attacks that take place over an unusually long period of time. This is a challenge for stateful detection of activity given finite time and memory resources. Operating slow and subtle is a deliberate strategy for some attackers to exhaust these resources and avoid being caught. Staniford et al propose an approach to detect traffic that is statistically anomalous from normal traffic. 22 Packets identified as anomalous are grouped with other packets (based on heuristics developed from real scans) and stored for a limited period of time. If a correlation is found between the packets in a group then a scan is detected.

Traffic scalability is essentially to do with monitoring high-speed links processing large volumes of data. This brings with it performance concerns, due to the latency introduced, and dropped packets due to increased highspeed traffic. This is in addition to issues to do with volume and resources mentioned in the earlier two cases. Efforts in this direction include work by Kruegel et al, who propose a traffic slicing approach to perform stateful detection for highspeed links, and Sekar et al who propose an efficient pattern matching algorithm that promises to support detection at up to 500 Mbps but only demonstrate it for offline traffic log files.15, 19

While these efforts contribute to addressing individual problems, they do not promise a coherent research direction. We propose to consider the problem from a system perspective that includes — strategies to make use of data collection mechanisms more efficiently, a serious rethink of the type of data analysis required, more succinct and more manageable, and — a realistic assessment of current sensor capabilities, and the costs incurred to deploy them, ultimately allowing us to use them optimally.

Challenges ahead

We are striving for a system-smart view of the scalability problem where the above essentially serve as broader design principles that cut across the three scalability themes mentioned earlier. Our message is simple: we need to be conscious of the need for monitoring and intrusion detection at every level of the system from the bottom up.

We layout our view on some of the scalability challenges that intrusion detection faces. We organise it in three main sections:
First we discuss the need for more efficient data collection approaches. Second, we propose a fundamental shift in data analysis from singular event detection to monitoring of behavioural characteristics. Finally, we focus on optimal sensor placement strategies.

All along, we highlight related work hitherto in respective areas. Towards the end, we also comment on how critical is good hardware support and highlight some relevant work in this area.

**Efficient Data Collection**

The cost of data collection in large networks is very high. As networks grow, along with the variety of components monitored, the cost of collecting data also grows. If this cost is to be minimised then the design and location of data collection mechanisms require a serious rethink.

One promising approach is to internalise sensors so that they are embedded in the code of the components being monitored, such as applications, operating systems and network infrastructure.14

So, for example, to monitor process information, if the operating system kernel is modified to log this information then the location of the sensor is internal (as it is part of the code); whereas if a separate process is running on the host and uses the ps command (on a Unix system) to retrieve this information then the process is considered external to the component being monitored.

Placing sensors internally in this sense provides numerous advantages in terms of accuracy and completeness, minimal delay between generation and reporting of data, and tamperproofness. But most importantly, it provides low computational overheads and also a lower cost of deployment.

Since the sensor process becomes part of the monitored component itself, the need for extra computational capacity is avoided. The cost of any deployment, including a separate platform to host the sensor process, external to a component is also done away with. Admittedly this is hard and initially expensive as it may even require modification of existing components.

Once achieved however, it promises an overall reduction of costs due to economies of scale. The important question is how can such initial costs be minimised?

We propose that such an approach is adopted gradually and opportunistically. The industry would need to be incentivised towards the redesign of software and hardware components for this purpose.

This is not entirely infeasible given the current political and technological climate where surveillance and monitoring is becoming norm.10 Open source software would certainly prove to be easier to adapt for such purposes.

Moreover, such an approach should not ignore using existing sources of data. Monitoring for network usage and management, for example, can be tapped into for the purposes of intrusion detection. Such sources provide data essentially as a by-product of existing collection mechanisms and therefore could provide very cheap means to collect data for intrusion detection.

**Towards behavioural analysis**

A critical part of intrusion detection is analysing the collected data. Be it individual packets from a network or process information from a host, good data analysis is important if difficult attacks are to be detected, and false positives and negatives also be minimised.

Such analysis have typically included (and in some cases are a combination of) signature matching, anomaly detection or some form of rule-based detection; other less common analyses include the use of artificial intelligence techniques and data mining (see Bace for a comprehensive survey on such techniques).3

Analysis of huge volumes of data, associated with an increased number of nodes and some of which potentially requiring stateful inspection over long periods of time, offer challenges in terms of both network and temporal scalability.

- **We propose a fundamental shift to**
  - **more emphasis on behaviour than data:** moving away from storing single events (such as packets, individual login attempts) and moving towards an assessment of the behaviour that such events represent. This can be achieved by:
    - Employing effective stateless analysis, and where not possible.
    - Maintaining a minimal summary of state, ideally with a view to summarising the behaviour of nodes rather than characterising individual packets or streams of traffic.
    - Exploring opportunities for data reduction to discard unwanted information, and as a result of all this,
    - More storage of useful information and less of unnecessary data.

To achieve the above stated goals, we need to turn the essential problem of data analysis on its head: what information do we need to store and analyse? Very little research has considered this problem. González et al present a hardware-based approach to characterise likely uninteresting traffic more cheaply that can then be devoid of further more expensive software-based analysis.11 They demonstrate their approach for potential Gigabit Ethernet operations.

"We need to turn the essential problem of data analysis on its head"

One approach to effective behavioural analysis, for example, is to move away the focus from attacks and toward attackers. Given a suspicious event of interest, we determine nodes that could be potentially responsible. Following more suspicious events, further nodes are identified in a similar fashion. Intersecting this set of nodes may indicate common nodes that warrant further investigation.

Such an approach could be implemented by simply accumulating a ‘score’ for each node in the system. The ‘score’ could be in the form of a count of the number of related events, with an alarm threshold (other alternatives could be some form of weight calculation or probability estimates). This means sufficient long-term state could be maintained given that the state size is a small multiple of the number of nodes and the need for storing individual events is done away with.
Restricting long-term state to essentially an incremental estimate of the probability that each node is subverted by an attacker offers both network and temporal scalability. For an approach essentially designed to identify nodes for further investigation, it allows to focus precious monitoring resources on a likely smaller, manageable set of nodes.

The authors are currently involved in implementing and testing such an approach. Early results are very promising.

Following down such a path we are likely to face a compromise of the granularity of such analysis and even accuracy. How much data can be sacrificed before loosing valuable information about the suspicious activity being detected? Essentially then, what is the right balance in terms of data discarded and retained for accurate analysis?

Optimal sensor deployments

Devising efficient deployment of intrusion detection sensors that maximise the visibility of suspicious events at a minimum cost offers a real challenge. With a possibly variable location and typically a range of configuration and analysis options, along with the possibilities to respond to potential intrusions, such sensors offer a range of intricate issues that need to be examined if one is to achieve an optimal deployment.

This raises the question: how do we achieve systematic sensor placement for modern complex networks? Of the related work in this area the most notable approach is attack graphs that represent an intruder’s actions as nodes and the changes caused in the network state as arcs.13, 26, 27, 28, 29

Given a starting node, a location in a network, an attack graph represents a series of states that the intruder could potentially traverse through to reach a final goal state. Such graphs could be useful to determine network security both in terms of a minimum set of critical locations to protect in a network to avoid most or all attacks, and the likelihood of successful attacks given particular starting nodes. While attack graphs are useful to characterise networks in terms of vulnerabilities, they do not scale up for large networks.16 More recent work shows more promise however.12, 30, 31

Still the analysis and response capabilities of sensors, and costs involved in deploying and using them to monitor networks are ignored.

For sensor deployments to become more system-smart, we propose a realistic assessment of:

- The capabilities of the current generation of intrusion detection sensors,
- Disruption costs due to the deployment of such sensors,
- Manual engagement required due to a high load of monitoring, and
- Characteristics of the network monitored.

Only a practical, system-level perspective on the above aspects can allow us to devise strategies to maximise the value achieved from a given set of sensors. Our attempt to characterise intrusion detection sensors for a range of detection abilities and deployment costs represents a first step towards this.21

We further propose emphasis on risk informed strategies to deploy sensors where they could be most effective to protect most valuable assets. Chivers et al introduce risk profiles for system components to characterise the risks to which a system is exposed if the component is compromised.5

The notion could be applied to network nodes to denote the level of risk exposure for the network if particular nodes are compromised. Such a calculation takes into account the value of a node as an asset, its location and the type of access it provides to penetrate further in a network, and the likelihood of intrusions targeting it.

Risk profiles serve to highlight, for example, that web servers, critical to the operation of an organisation engaged in electronic commerce and likely to have more access to critical information assets, are typically at a higher risk than ordinary client nodes residing on some internal segment; the higher the risk the more the need to monitor such nodes. Such an approach allows risk assessment to be carried out in an incremental and distributed fashion, yet offering convergence on a single consistent risk profile for the entire system. This provides significant advantages over vulnerability- driven approaches, such as attack graphs, which evidently fall short on the scalability front.

The authors have built on this work to propose a deployment value model for intrusion detection sensors.20 Our representation of various location (risk profile, load) and sensor (capability, efficiency, cost) characteristics allows us to determine the value of sensors operating at particular locations in a deployment. The risk profile is important as it identifies how crucial are locations for the purposes of monitoring in mitigating risks to the network. In that sense the higher the capability deployed to mitigate the maximum risk, the higher the value of a deployment.

We also propose a deployment strategy the purpose of which is to maximise the deployment value of a set of sensors placed over a set of locations. The ideal strategy would attempt to ensure that the total value of deployed sensors is maximal and hence fair. Implementation of the model employs a simple exhaustive search for the highest deployment value for every available sensor over a compatible location, computing the total deployment value cumulatively with 1000 sensor deployments being computed in a few minutes.

Further work aims to use other search strategies to optimise the calculation further; initial results are very encouraging.4 A highly scalable, risk-based approach to deploy multiple types of sensors in a systematic fashion could provide a significant contribution for intrusion detection on large scale networks.

Hardware support

Network traffic capture on high-speed links is always a challenge due to capacity issues. This results in often significant packet loss. To cope with the ever-rising speed of links, there is a need to re-examine how such traffic is collected and preprocessed for further analysis.

One promising approach is offered by Watson and Peron, who have revised the memory model used by intrusion detection processes to analyse packets.24 They
have implemented their approach for the Berkeley Packet Filter (BPF) in the FreeBSD operating system, where traditionally a user process interface uses a system call to read packets from a kernel buffer. This means the packet is copied twice, first from the device driver to the kernel buffer and secondly from the kernel to user memory. This incurs a significant performance overhead alongside the possibility of the kernel overwriting the packets while they are being analysed and hence packet loss. The revised approach presents a zero-copy buffer solution where packets are written directly onto a memory buffer shared between the BPF and the intrusion detection process. This avoids both system call and copying overhead.

“Network traffic capture on high-speed links is always a challenge due to capacity issues”

Such approaches demonstrate how existing system architectures need to be reconsidered to make efficient use of hardware. González et al’s work discussed earlier is also worthy of note here.11

Paxson et al bring to light the shortcomings of traditional use of hardware for this purpose and propose efficient exploitation of parallelism if hardware approaches are to provide a substantial contribution for this purpose.16 Approaches to estimating hardware resource consumption over high volume high-speed links are also useful here.8

Discussion

Intrusion detection sensor deployments also need to cope with the increasing disparate nature of the composition of modern networks: some parts may be very different to others in terms of the topology, the traffic they encounter, the kind of threats they are faced with, and hence the kind of intrusion detection they need. As parts of a network evolve, in terms of traffic and topology, possibly dynamically, how are the existing and any additional resources best deployed? The question essentially here is whether it is possible to perform intrusion detection in a functionally scalable way? So, for example, a different set of intrusion signatures are used for different parts of the network, or a different type of analysis entirely. Data collection mechanisms may also need to change. All of this may also require a more localised response strategy. The challenge is how do we design distributed monitoring and detection mechanisms which act independently and efficiently, but also coherently towards the goal of protecting diverse yet a single entity?

Larson et al present a first step towards such a capability.25 They propose a new architecture for a monitoring infrastructure that allows adaptive intrusion detection: detection algorithms can be selectively enabled (or disabled) and sensors can be reconfigured to collect relevant data as detection and resource needs dictate. The architecture is designed to be operator-centric, which means it is driven largely manually and any adaptation is based on human observation. This is early work and Larson et al discuss this only at a conceptual level.25

The purpose of our effort is to adopt a systems perspective towards scalable intrusion detection. Efficient data collection requires a serious rethink of the design of intrusion detection sensors. If deployment and operating costs are to be lowered, then internalising with respect to monitored components would become ever more necessary. Managing the data collected, in terms of recognising threats and prioritising response, for large scale deployments is an even bigger challenge.

Summative and reductive approaches to analysing such data allow efficient detection, especially for time dispersed activity. Optimal placement of sensors, often limited in number, on ever growing large networks presents another potent problem. As monitoring costs grow, relative to detection infrastructure resources, it becomes imperative to spend these resources more efficiently.

Security is a tradeoff

The case with intrusion detection is no different. When it comes to deploying scalable intrusion detection such a tradeoff only becomes more visible. The question is how do we make such a tradeoff more tenable?

Acknowledgment

This effort is a result of a collaborative project between Cranfield University and the University of York funded by the EPSRC (EP/E028268/1) to study system-level approaches to intrusion detection. Some of the ideas presented have benefitted from discussions with Juan M. Estévez Tapiador.

References

3. Rebecca Bace. Chapter 4. analysis schemes.