# Collaborative Intrusion Detection Networks and Insider Attacks

Carol Fung
*University of Waterloo*
Waterloo, ON, Canada
j22fung@uwaterloo.ca

**Abstract**

Cyber intrusion is becoming an increasingly global and urgent problem. Intrusion Detection Systems (IDSs) are deployed to identify intrusions and mitigate their damage. A stand alone IDS does not have complete information or knowledge to detect intrusions. A Collaborative Intrusion Detection Network (CIDN) consists in a set of cooperating IDSs which use collective knowledge and experience to achieve improved intrusion detection accuracy. However, insider attackers may severely degrade the efficiency of CIDNs. This paper provides a survey of some CIDNs and analyzes their robustness against insider attacks. We first classify network intrusions, IDSs, and insider attacks for CIDNs according to their behaviors and the techniques they use. A taxonomy of CIDNs is then provided with an analysis based on criteria of topology, scope, specialization, data privacy awareness, and their vulnerabilities to insider attacks. Some of the open challenges and future directions in cooperative CIDNs are discussed in the last section.

## 1 Introduction

In modern days almost all computers are connected to the Internet. Applications which rely on networks such as email, web-browsing, social networks, remote connections, and online chatting are being used by billions of users every day. At the same time, network intrusions are becoming a severe threat to the privacy and safety of computer users. By definition, network intrusions are unwanted traffic or computer activities that may be malicious and destructive. The consequence of a network intrusion can be the degradation or termination of the system (denial of service), user identity information theft (ID theft), unsubscribed commercial emails (spam), or fraud of legitimate websites to obtain sensitive information from users (Phishing). Network intrusions usually rely on the executing of malicious code (Malware) to achieve their illegal goals. People who write or control malicious code are called *hackers*. In recent years, network intrusions are becoming more sophisticated and organized. Intruders tend to control a group of compromised computers/hosts to launch distributed attacks, for example, Distributed Denial of Service (DDOS) attack. Compromised Bot nodes, which may run on Malware, communicate with a Bot master through a command and control server, HTTP, or peer-to-peer communication network [1]. The group of compromised nodes with a master together forms a Botnet. "Bot nodes" can be used to commit profit-driven cyber crimes such as DDOS attacks, spam spreading, ID theft, or Phishing.

To protect computer users from the damage of malicious intrusions, intrusion detection systems can be used to detect illegal traffic or computer activities and further alert the system administrator to take corresponding actions. Intrusion Detection Systems (IDS) are software/hardware designed to monitor network traffic or computer activities and alert administrators for suspicious intrusions. Based on the model of detection, IDSs can be divided into signature-based and anomaly-based. Based on data sources, they can be host-based or network-based. Signature-based (misuse) IDSs compare data packets with the signatures or attributes database of known intrusions to decide whether the observed traffic is malicious or not. Anomaly-based IDSs observe traffic or computer activities and detect intrusions by identifying activities distinct from a user's or a system's normal behavior. A Host-based IDS (HIDS) runs on an individual host or device in the network (Figure 1). It monitors inbound/outbound traffic to/from a computer as well as the internal activities such as system calls. Some examples of HIDSs are OSSEC [2] and tripwire [3]. Network-based IDSs (NIDS) monitor network traffic packets, such as TCP-dump, to/from the network system. Examples of NIDSs are Snort [4] and Bro [5].

Isolated intrusion detection systems do not have information about the whole environment and may be easily compromised by new intrusions. Cooperation in intrusion detection systems enables the system to use collective information from other IDSs to provide more accurate intrusion detection locally or system wide. The network that connects IDSs to exchange information among them is a cooperative intrusion detection network (CIDN). However, attackers may compromise some peers in the CIDN and use the compromised peers to attack the collaboration system, in order to invade the system by compromising the defense capability of the network. Malicious peers can make use of some common attacks, such as Sybil attacks, newcomer attacks, betrayal attacks, and Collusion attacks, to degrade the efficiency of the CIDN by sending false information, out-voting honest nodes, or even attacking and compromising some honest nodes in the CIDN. Therefore, a robustness design against insider attacks is very important to the CIDN. In this paper, we provide a literature survey of existing CIDN models and analyze their properties, such as topology, scope, specialization, data privacy awareness, and robustness to various insider attacks.

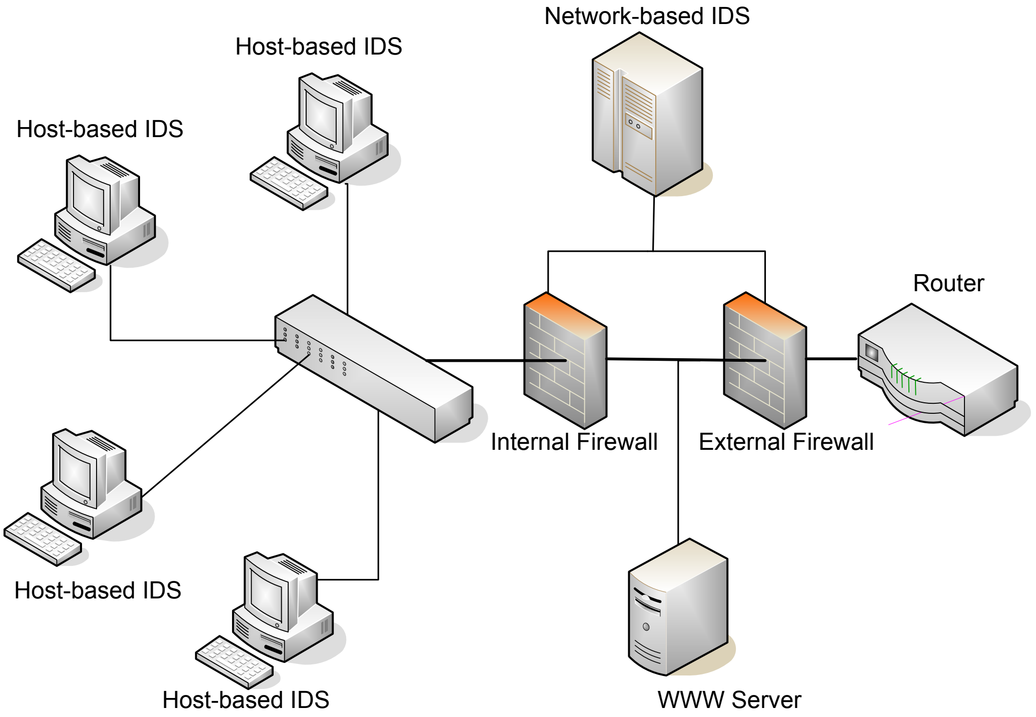


Figure 1: An example of host-based IDS and Network-based IDS

The rest of the paper is organized in the following: In Section 2, we describe intrusion detection networks (IDN) and several criteria to classify them. We list some common insider attacks to CIDNs in Section 3. In section 4, we briefly describe and analyze a list of CIDNs in literature and provide a taxonomy to classify them according to our criteria. Finally, we predict some open challenges in Section 5 and summarize this paper in Section 6.

# 2 Cooperation in Intrusion Detection Networks

Many different CIDNs have been proposed in the past a few years. Based on our knowledge, most CIDNs collect data from distributed peer IDSs and to enable more accurate intrusion detection. In this section, we provide a set of criteria (i.e. cooperation topology, cooperation scope, and specialization) to categorize CIDNs. A taxonomy of CIDNs is then built using these criteria we propose.

## 2.1 Cooperation Topology

The cooperation topology of a CIDN can be centralized or decentralized. In a centralized system, all the intrusion data from end nodes are forwarded to a central server for analyzing and processing. In general, a centralized system (e.g., DShield [6] and CRIM [7]) has the advantage of having complete data and can potentially provide more accurate detection. However, the disadvantage is that it may cause traffic clog close to the analyzing center and the server is a single point of failure. On the other side, in a decentralized system, intrusion data are sent to different places for processing and analyzing. A decentralized system can be fully distributed or partially decentralized. In a fully distributed system (e.g., Indra [8], NetShield[9], and HBCIDS[10]), all nodes in the network play equal roles in cooperation as both data contributors and analyzers. The failure of a single node will have little impact on the functionality of the cooperation network. However, the lack of full data in each analyzer may lead to less accurate intrusion detection. In a partially decentralized system, some nodes may take the responsibility of analyzing data, and therefore, have heavier workload than peers which only contribute with data. The network structure may be clustered (e.g. ABDIAS[11]) or hierarchical (e.g. DOMINO[12]). A partially decentralized system targets to find a balanced solution between the centralized system and the fully distributed system.

## 2.2 Cooperation Scope

Another feature which can be used to categorize CIDNs is the cooperation scope. The cooperation scope of a CIDN can be local, global, or hybrid. In a local-scope CIDN (e.g. Indra[8] and Gossip[13]), peers in the CIDN are usually assumed to be fully trusted. The privacy concern of exchanging packet payload is usually neglected since all nodes lie in the same administrative boundary. Therefore, data packets can be in full disclosure and exchanged freely among peers. In a global CIDN (e.g. DShield [6] and NetShield [9]), peers exchange intrusion information with other IDSs outside administration boundaries. Therefore only limited information can be shared since privacy is a concern. In this case, data payload (or IP addresses, etc.) is either digested or removed in the exchanged information. In a hybrid system (e.g. DOMINO[12] and ABDIAS[11], the network is divided into different trust zones. Different data privacy policies are applied inside different zones depending on the level of trust inside the zone.

## 2.3 Specialization

An CIDN can be dedicated to a specific intrusion such as worms (e.g. NetShield[9], Gossip[13] and Worminator[14]), spam (e.g. ALPACA[15]), or can be used to detect general intrusions (e.g. Indra [8], CRIM [7], and HBCIDS [10]).

## 2.4 Data Privacy Awareness

Data exchanged among IDSs may leak private information of users or their networks to unauthorized parties. Malicious insiders can take advantage of collected private information such as IP addresses and port numbers of participating nodes for illegal usage. Some strategies of preserving data privacy or reducing information leak in a CIDN include data filtering across trust zones (e.g., DOMINO [12]), exchanging only digested data (e.g., NetShield [9] and Worminator [14]), and removing sensitive information from the exchanged data (e.g., DShield [6]).

# 3 Insider Attacks on Intrusion Detection Networks

Collaboration among IDSs can effectively improve the efficiency of intrusion detection. However, the IDS network itself may become the target of attacks and be compromised. The traditional communication attacks such as eavesdropping, man-in-the-middle, replaying, and cloning can compromise the security of information exchanged inside a CIDN. Correspondingly, those attacks can be prevented by traditional solutions such as encryption/decryption and public/private key infrastructure. In this section, we focus on insider attacks, which are initiated by peers inside the collaboration network. Some common insider attacks to IDS networks are listed in the following.

## 3.1 Sybil attacks

A Sybil attack represents a situation in which a malicious peer in the system creates a large amount of pseudonyms (fake identities) [16]. This malicious peer uses fake identities to gain larger influence in the network. For example, fake nodes can out-vote honest nodes or send a lot of false information to render the collaboration network useless. The Sybil attacks are especially successful when the system registration is open to the public and authentication is loosely controlled. The lack of authentication mechanisms provides opportunities for malicious peers to create a large number of fake identities. Several common strategies can be used to throttle Sybil attacks. For example, strict authentication makes registering fake identities difficult; collaboration which requires human labors or expensive resource limits the number of fake identities a malicious peer can create.

## 3.2 Newcomer attacks

A newcomer attack occurs when a malicious peer registers as a new user after having a bad behavior record in the network [17]. Such a malicious peer creates a new ID for the purpose of erasing its bad history with other peers in the network. Newcomer attacks can degrade the efficiency of the collaboration and cause honest users to lose the benefit from honest behaviors. An effective CIDN model shall discourage newcomer attacks by minimizing the benefit from such behaviors. Defense strategies against newcomer attacks include lowering the impact from newcomers and using enforced probation periods for all new users.

## 3.3 Betrayal attacks

when a trusted peer suddenly turns into a malicious one and starts sending false information or even attacking the other peers, this is called a betrayal attack. A trust-based collaboration system can be degraded dramatically because of this type of attacks. A countermeasure against such attacks is to severely punish the betrayed nodes and nodes with inconsistent behaviors. A trust mechanism robust to betrayal attacks shall satisfy the social norm: “It takes a long-time interaction and consistent good behavior to build up a high trust, while only a few bad actions to ruin it.” When a trustworthy peer acts dishonestly, its trust value should drop down quickly, hence making it difficult for this peer to deceive others or gain back its previous trust within a short time.

### 3.4 Collusion attacks

Collusion attacks occur when a group of compromised/malicious peers cooperate together in order to compromise the network. A large number of colluded malicious nodes may out-vote honest nodes, lead the system to make false intrusion decisions, or compromise other honest nodes. The influence is usually proportional to the percentage of colluded malicious nodes in the network. The collusion attack is hard to eliminate but can be suppressed if nodes only use direct experience to evaluate the trustworthiness of their collaborators.

### 3.5 Hybrid attacks

Attackers can also use a mixture of different attacks in their strategies to gain higher impact, such as using the Sybil attack to create a large amount of fake identities and then use collusion attacks to achieve their goals. Other strategies include, but not limited to, betrayal attacks with newcomer attacks, and betrayal attacks with Collusion attacks.

## 4 Intrusion Detection Networks and Taxonomy

In this section, we briefly summarize and analyze a list of selected CIDNs. We classify them using several criteria such as topology, scope, specialization, and vulnerability to insider attacks. A taxonomy is provided in the last subsection to illustrate the classification.

### 4.1 Indra

Indra [8] is one of the early papers to propose a cooperative intrusion detection system (Figure 2). In the proposed system, host-based IDSs in a local area network take a pro-active approach and send warnings to other trusted nodes about the intruder through a peer-to-peer network. For example, as shown in Figure 2, if an attacker compromises a weak node B then launch attacks from B to hosts in the trusted network. Node C detects an attack from B and then multicasts a security warning to its trusted neighbors regarding B. Therefore, if B intends to attack other devices in the network, it will be repelled straight away by the forewarned nodes. Indra is a fully distributed system which targets on local area network. Indra is a local collaboration network with fixed participants, so Sybil attacks and newcomer attacks are not applicable. However, compromised nodes can easily send false alerts to mislead the other nodes to make incorrect intrusion decisions. Therefore, it is vulnerable to betrayal attacks and Collusion attacks.

### 4.2 DOMINO

DOMINO [12] is an IDS collaboration system which aims at monitoring Internet outbreaks for a large scale network. In the system architecture (Figure 3), heterogeneous IDSs located in diverse locations share their intrusion information with each other. There are typically three types of nodes, namely, axis nodes, satellite nodes, and terrestrial contributors. Satellite nodes are organized hierarchically and are responsible for gathering intrusion data and send them to parent nodes in the hierarchical tree. Parent nodes aggregate intrusion data and further forward data up to the tree till they reach axis nodes; Axis nodes analyze intrusion data, generate digested summary data and then multicast them to other axis nodes. Network-based IDSs and active sink nodes (such as Honeypot [18]) are integrated to axis nodes to monitor unused IP addresses for incoming worms; Terrestrial contributors do not follow DOMINO protocols but they can also contribute to the system through DOMINO access points. In DOMINO, heterogeneous nodes are involved in the cooperation overlay. Information from axis nodes, satellite nodes,

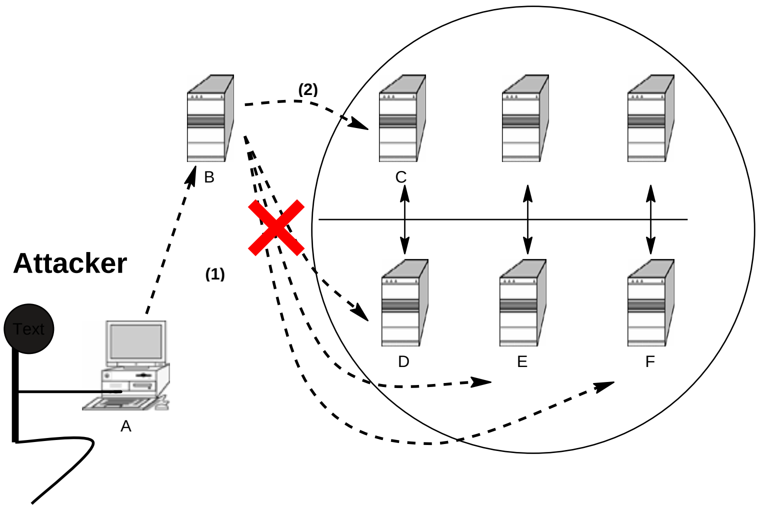


Figure 2: Indra Architecture (Adapted from [8])

and terrestrial contributors is distinguished by different trust levels. This feature enables DOMINO to handle inter-administration-zone cooperation. Sensitive data is filtered when sent to different trust zone. DOMINO is a decentralized system with hierarchical structure. It is a global CIDN with a good scalability. Regarding insider attacks, DOMINO has a filtering system to control the data flow from one or a small group of compromised nodes. However, it also employs a certificate authority to control fake identities. Therefore, DOMINO is robust to Sybil attacks and newcomer attacks. However, the lack of sophisticated trust/reputation management system makes it vulnerable to betrayal attacks and Collusion attacks.

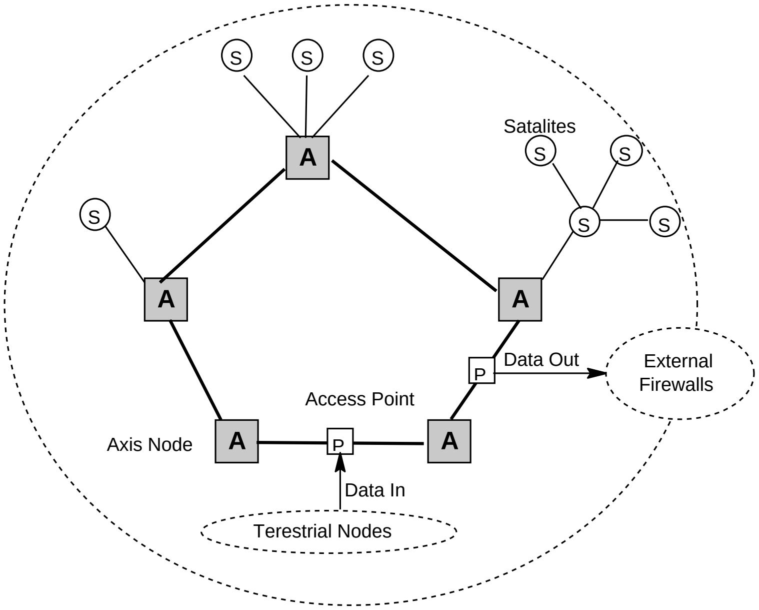


Figure 3: DOMINO Architecture (Adapted from [12])

## 4.3 DShield

DShield [6] is a community-based firewall log correlation system. The central server receives firewall logs from world-wide volunteers and then analyzes attack trends based on the information collected. Similar systems includes myNetWatchMan [19] and CAIDA [20]. DShield is used as the data collection engine behind the SANS Internet Storm Center (ISC) [21]. Analysis provided by DShield has been used in the early detection of several worms, such as "Code Red" and "SQL Snake". Due to the number of participants and volume of data collected, DShield is a very attractive resource and its data is used by researchers to analyze attack patterns. However, DShield is a centralized system and it does not provide real-time analysis or rule generation. Also due to the privacy issues, payload information and some headers can not be shared, which makes the classification of attacks often not possible. However, DShield does not require restrict authentication to participate and also does not have trust evaluation for all participants. Therefore, it is also vulnerable to all insider attacks listed.

## 4.4 NetShield

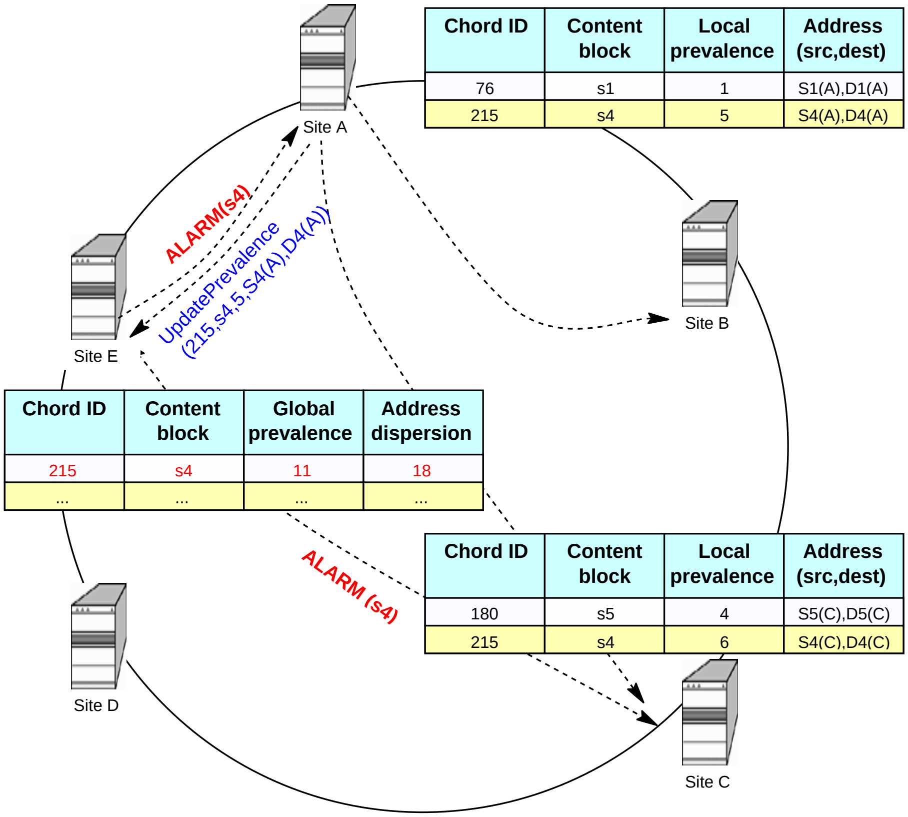
NetShield [9] is another CIDN which uses the DHT Chord system to reduce communication overhead. However, in the system architecture (Figure 4), IDSs contribute and retrieve information from the system through a P2P overlay. Each IDS maintains a local prevalence table to record the number of occurrences of each content block signature locally and its corresponding source address and destination address. An update will be triggered if the local prevalence of the content block exceeds a local threshold (for example, site A in Figure 4). If the global prevalence is higher than a threshold, and the address dispersion exceeds a certain threshold then an alarm will be raised regarding the corresponding content block. Netshield targets on epidemic worm outbreaks or DOS attacks. However, the limitation is that using content blocks as attack identification is not effective to polymorphic worms. Also the system assumes all participants are honest, which is vulnerable to the collusion attack since a group of malicious nodes can send false information to cause significant high false alarm rate.

## 4.5 Gossip

Denver et.al. [13] proposed a collaborative worm detection system for enterprise level CIDN for host-based IDSs. A fully distributed model is adopted to avoid a single point of failure. In their system, a host-based IDSs (local detector) raises an alert only if the number of newly created connections per unit time exceeds a certain threshold. The alert will then be propagated to neighbors for aggregation. A Bayesian network based alert aggregation model is used here for alert aggregation on global detectors. Their proposed system aims at detecting slow propagating worms in a Local area network. However, their system only uses new connection rate as the sign of possible worm spreading. This is not effective to worms spreading without connections, such as UDP worms. However, compromised nodes can easily cause the system to raise false alarms.

## 4.6 Worminator

Worminator [14] was proposed to enable IDSs to share alert information with each other to detect worm propagation. Alert correlation is used to gain better detection accuracy. Different from most other systems, Worminator addresses concerns about the privacy of exchanging alerts and proposed to use bloom filter to encode IP addresses and port numbers in the alerts to preserve privacy of collaborators. Worminator claims that the system topology can be either centralized or decentralized depending on the size of the network. Worminator is a system open to the public organizations. The lack of sophisticated trust management system makes it vulnerable to various insider attacks.



| Chord ID | Content block | Local prevalence | Address (src,dest) |
|---|---|---|---|
| 76 | s1 | 1 | S1(A),D1(A) |
| 215 | s4 | 5 | S4(A),D4(A) |

| Chord ID | Content block | Global prevalence | Address dispersion |
|---|---|---|---|
| 215 | s4 | 11 | 18 |
| ... | ... | ... | ... |

| Chord ID | Content block | Local prevalence | Address (src,dest) |
|---|---|---|---|
| 180 | s5 | 4 | S5(C),D5(C) |
| 215 | s4 | 6 | S4(C),D4(C) |

Figure 4: NetShield Architecture (Adapted from [9])

## 4.7 ABDIAS

Ghosh et al. proposed an Agent-Based Distributed Intrusion Alert System (ABDIAS) [11]. In the architecture design (Figure 5), IDSs (agents) are grouped into communities (neighborhoods). Each Agent collects information inside its neighborhood and uses a Baysian network analysis model to diagnose for possible threats. Inter-neighborhood communication only happens if consensus can not be reached within a neighborhood. This system supports early warnings for pre-attack activities to gain time for administrators to respond to potential attacks. This system also supports a simple majority-based voting system to detect compromised nodes. However, a voting-based system is vulnerable to collusion attacks.

## 4.8 CRIM

CRIM [7] is a cooperative IDS where alerts from individual IDSs are sent to a central analyzer for clustering and correlating. A set of correlation rules are generated offline by security administrators by analyzing attack descriptions. These correlation rules are then used for analyzing alerts collected from IDSs to recognize global attack scenarios. CRIM is a semi-automatic alert correlation system since it relies on human interactions to define attack descriptions. It is also a centralized system. CRIM is vulnerable to compromised insiders since false messages from insiders may mislead the system to generate false alarms.

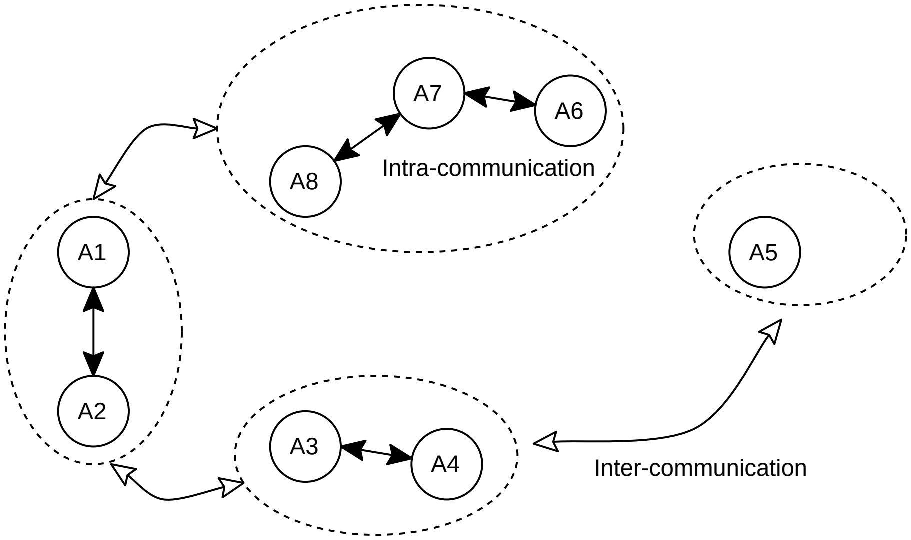


Figure 5: ABDIAS Architecture (Adapted from [11])

## 4.9 HBCIDS

Fung et al. proposed a Host-based Collaborative Intrusion Detection System (HBCIDS) [10], where host-based IDSs are connected following a social network pattern. Each HIDS has a list of acquaintances with which they exchange alert/attack information to request immediate diagnosis. [10] uses a simple trust management system to distinguish honest nodes and dishonest nodes, thereby improving the efficiency of the collaboration. IDSs use test messages (e.g. attack traffic or legitimate traffic where their outcomes are known before-hand) to evaluate the trustworthiness of other nodes. In the case of intrusion detection, the weight of the diagnosis feedback from an acquaintance is proportional to its corresponding trust value. [22] further proposed a Bayesian statistic model to trace the confidence level of trust estimation. The confidence level is then used to find the lowest test messages rate to guarantee trust estimation confidence. This collaboration system is a fully distributed system. HBCIDS has admission control and sophisticated trust management system. Direct experience is used to evaluate the trustworthiness of collaborators. Therefore, the influence of Collusion attacks is controlled in this CIDN.

## 4.10 ALPACAS

[15] is a cooperative spam filtering system that aims at preserving the privacy of emails as well as gaining scalability of the system. The system is built on top of a peer-to-peer overlay to avoid the deficiency of a centralized system. Spam mails and ham mails are distributed on agents based on the range of their feature signatures. An email is divided into feature trunks and trunks are digested into feature finger prints to preserve content privacy of emails. Figure finger prints of an email are then sent to corresponding agents to do comparison with stored spam emails and ham emails by estimating the maximum signature overlap with spam (MOS) and the maximum signature overlap with ham (MOH). An email is a labeled as a spam if the difference between MOS and MOH exceeds a certain threshold. ALPACAS is a fully distributed system without authentication system specified. ALPACAS is based on the assumption that all participants are trustworthy. Compromised insiders can send incorrect spam and ham mails to compromise the detectability of the whole system. Therefore, It is vulnerable to all insider attacks listed.

Table 1: Classification of Cooperative Intrusion Detection Networks

| CIDN | Topology | Scope | Specia-lization | Privacy Awareness | Sybil Attack | Newcomer Attack | Betrayal Attack | Collusion Attack |
|---|---|---|---|---|---|---|---|---|
| Indra | Distributed | Local | Worm | N | - | - | Y | Y |
| DOMINO | Decentralized | Hybrid | Worm | Y | N | N | Y | Y |
| DShield | Centralized | Global | General | Y | Y | Y | Y | Y |
| NetShield | Distributed | Global | Worm | Y | Y | Y | Y | Y |
| Gossip | Distributed | Local | Worm | N | - | - | Y | Y |
| Worminator | Any | Global | Worm | Y | Y | Y | Y | Y |
| ABDIAS | Decentralized | Hybrid | General | N | Y | Y | N | Y |
| CRIM | Centralized | Local | General | N | - | - | Y | Y |
| HBCIDS | Distributed | Global | General | N | N | N | N | N |
| ALPACAS | Distributed | Global | Spam | Y | Y | Y | Y | Y |

### 4.11 Taxonomy

Based on the criteria we provide above, we categorize a list of selected CIDNs and illustrate it using a taxonomy as in Table 1. The criteria used are topology, scope, and specialization. The data privacy protection and vulnerabilities of each CIDN to various insiders attacks are also listed. The sign "-" indicates that the attack is not applicable or there is not enough information to make a judgment. The sign "Y" represents data privacy aware or vulnerable to an attack. The sign "N" means no data privacy protection or not vulnerable to an attack.

We can see that local-scope CIDNs can avoid Sybil attacks and newcomer attacks. This is because most local-scope CIDNs have fixed list of participants, which makes the creation of fake IDs difficult. We also notice that all local-scope CIDNs do not provide data privacy protection. This is because data sharing in local computers are not necessary in general since they are in the same administration zone. Several global/hybrid-scope CIDNs provide data privacy protection. For example, DOMINO divides the network into different trust zones and sensitive data is filtered when sent to less trusted zones. Among all CIDNs listed above, both ABDIAS and HBCIDN can detect betrayal attacks since they have trust evaluation system to differentiate honest nodes and dishonest nodes. However, only HBCIDN is robust to collusion attacks. This is because ABDIAS uses voting mechanism for trust evaluation, which is vulnerable to collusions. HBCIDN nodes use direct experience to evaluate the trustworthiness of others. Overall, not a single CIDN is perfect to both data privacy reservation and insider attacks. Therefore, we recommend decision makers to leverage the priority of their desired features of their CIDN to choose an appropriate model.

## 5 Open Challenges and Future Directions

Although many CIDNs have been proposed and built, there are still many challenges mainly related to the collaboration efficiency. The first challenge is the privacy of exchanged information. Data privacy is a critical issue that discourage users from joining collaborations. This is especially true when the collaboration crosses administrative boundaries. Although some existing works propose to digest sensitive information in exchanged data such as IP addresses and port numbers to prevent privacy leaking [14].

However, to have data privacy while keeping efficiency of collaboration is still an open challenge. The second challenge of CIDNs is the incentive design. An incentive compatible collaboration system is a key component for the long term sustainability of all collaborations.

Using a CIDN to detect Botnets is a new but urgent open challenge. Botnets are getting more sophisticated and stealthy with time. Recent Peer-to-peer Botnets [23] are even more difficult to detect than traditional centralized Botnets. How to detect and remove Botnets will be a hot topic in the next few years. A few recent papers [24] [25] [26] have appeared to address how to detect fast-flux Botnet. [26] shows that collaborative detection is more efficient than single point detection. We are expecting more work to appear in this topic.

## 6 Conclusion

Collaborative Intrusion Detection Networks enable member IDSs to make use of the collected information or experience to enhance intrusion detection accuracy. CIDNs can be categorized using many different criteria. In this paper we summarized a list of existing CIDNs and categorize them according to their topology, scope, and specialization. We also analyze their data privacy awareness and robustness against common insider attacks, such as Sybil attacks, newcomer attacks, betrayal attacks, and Collusion attacks. There is a pressing need for the research community to develop common metrics and benchmarks for CIDN evaluation. The taxonomy will be helpful in shaping these tasks and may be enriched with the appearance of new CIDNs.

## References

[1] Y. C. Z. F. P. R. Z. Zhu, G. Lu and K. Han, “Botnet research survey,” in *Proc. of the 32nd Annual IEEE International Conference on Computer Software and Applications (COMPSAC’08), Turku, Finland*. IEEE, July-August 2008, pp. 967–972.

[2] “OSSEC,” http://www.ossec.net, Accssed January 2011.

[3] “TripWire,” http://www.tripwire.com, Accssed January 2011.

[4] “Snort,” http://www.snort.org, Accssed January 2011.

[5] “Bro,” http://www.bro-ids.org, Accssed January 2011.

[6] J. Ullrich, “DShield,” http://www.dshield.org/indexd.html, Accssed January 2011.

[7] F. Cuppens and A. Miege, “Alert correlation in a cooperative intrusion detection framework,” in *Proc. of 2002 IEEE Symposium on Security and Privacy, Oakland, California, USA*. IEEE, May 2002, pp. 202–215.

[8] R. Janakiraman and M. Zhang, “Indra: a peer-to-peer approach to network intrusion detection and prevention,” in *Proc. of the 12th IEEE International Workshops on Enabling Technologies (WETICE’03), Linz, Austria*. IEEE, June 2003, pp. 226–231.

[9] M. Cai, K. Hwang, Y. Kwok, S. Song, and Y. Chen, “Collaborative internet worm containment,” *IEEE Security & Privacy*, vol. 3, no. 3, pp. 25–33, 2005.

[10] C. Fung, O. Baysal, J. Zhang, I. Aib, and R. Boutaba, “Trust management for host-based collaborative intrusion detection,” in *Proc. of the 19th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management (DSOM’08), Samos Island, Greece*. IEEE, September 2008.

[11] A. Ghosh and S. Sen, “Agent-based distributed intrusion alert system,” in *Proc. of the 6th International Workshop on Distributed Computing (IWDC’04), Hiroshima, Japan, LNCS*, vol. 3326. Springer-Verlag, November 2004, pp. 240–251.

[12] V. Yegneswaran, P. Barford, and S. Jha, “Global intrusion detection in the domino overlay system,” in *Proc. of Network and Distributed System Security Symposium (NDSS’04), San Diego, USA*, February 2004.

[13] D. Dash, B. Kveton, J. Agosta, E. Schooler, J. Chandrashekar, A. Bachrach, and A. Newman, “When gossip is good: Distributed probabilistic inference for detection of slow network intrusions,” in *Proc. of the National*

*Conference on Artificial Intelligence*, vol. 21, no. 2. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006, p. 1115.

[14] A. K. M.E. Locasto, J.J. Parekh and S. Stolfo, "Towards collaborative security and p2p intrusion detection," in *Proc. of 2005 IEEE Information Assurance Workshop (IAW'05), New York, USA*. IEEE, June 2005, pp. 30–36.

[15] Z. Zhong, L. Ramaswamy, and K. Li, "Alpacas: A large-scale privacy-aware collaborative anti-spam system," in *Proc. of IEEE INFOCOM 2008, Phoenix, Arizona, USA*. IEEE, June 2008, pp. 556–564.

[16] J. Douceur, "The sybil attack," in *Proc. of the 1st International Workshop on Peer-to-Peer Systems (IPTPS'02), Cambridge, MA, USA, LNCS*, vol. 2429. Springer-Verlag, March 2002, pp. 251–260.

[17] P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman, "Reputation systems," *Commun. ACM*, vol. 43, no. 12, pp. 45–48, 2000.

[18] D. Dagon, X. Qin, G. Gu, W. Lee, J. Grizzard, J. Levine, and H. Owen, "Honeystat: Local worm detection using honeypots," in *Proc. of the 7th International Symposium on Recent Advances in Intrusion Detection (RAID'04), French Riviera, France, LNCS*. Springer-Verlag, September 2004, pp. 39–58.

[19] "myNetWatchman," http://www.mynetwatchman.com, Accssed January 2011.

[20] "CAIDA: The Cooperative Association for Internet Data Analysis," http://www.caida.org, Accssed January 2011.

[21] "SANS Internet Storm Center (ISC)," http://isc.sans.org, Accssed January 2011.

[22] C. Fung, J. Zhang, I. Aib, and R. Boutaba, "Robust and sscalable trust management for collaborative intrusion detection," in *Proc. of the 11th IFIP/IEEE International Symposium on Integrated Network Management (IM'09), Long Island, NY, USA*. IEEE, June 2009.

[23] J. Grizzard, V. Sharma, C. Nunnery, B. Kang, and D. Dagon, "Peer-to-peer botnets: Overview and case study," in *Proc. of the First USENIX Workshop on Hot Topics in Understanding Botnets (HotBots'07), Cambridge, MA, USA*, August 2007.

[24] T. Holz, C. Gorecki, K. Rieck, and F. Freiling, "Detection and mitigation of fast-flux service networks," in *Proc. of the 15th Annual Network and Distributed System Security Symposium (NDSS'08), San Diego, USA*, February 2008.

[25] C. V. Zhou, C. Leckie, S. Karunasekera, and T. Peng, "A self-healing, self-protecting collaborative intrusion detection architecture to trace-back fast-flux phishing domains," in *Proc. of the 2nd IEEE Workshop on Autonomic Communication and Network Management (ACNM'08), Salvador, Brazil*. IEEE, April 2008.

[26] C. V. Zhou, C. Leckie, and S. Karunasekera, "Collaborative detection of fast flux phishing domains," *Journal of Networks*, vol. 4, no. 1, pp. 75–84, February 2009.

**Carol Fung** is a PhD student in University of Waterloo (Canada). She received her BSc and Msc from University of Manitoba, Canada. Her research topic is collaborative Intrusion Detection networks, which includes trust management, collaborative decision, resource management, and incentive design of such a system. She is also interested in the security problems in wireless networks and social networks.