การตรวจจับ Botnet แบบ RaelTime

โดยใช้ DNS-based และ Mining based

Real-time Botnet Detection using DNS-based and Mining based

ทศพร จันทรสุกรี, ชัชวาลย์ มุ่งแสง, ธวัชชัย เรืองธนานุรักษ์

สาขาเทคโนโลยีสารสนเทศ ภาควิชาวิทยาศาสตร์คอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยขอนแก่น

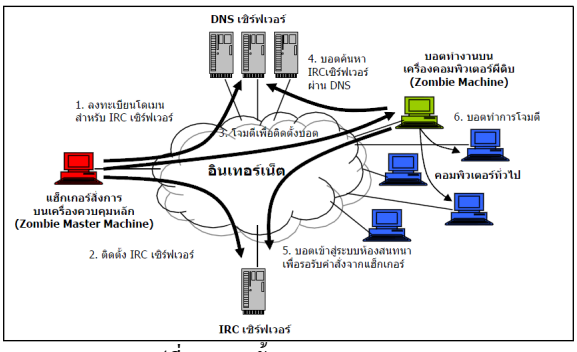
E-mail: todsapron@gmail.com, chatchawan.m@gmail.com, firekeenjoe@gmail.com

บทคัดย่อ— การตรวจสอบ Botnet มีวิธีการตรวจสอบไม่กี่วิธี การตรวจสอบแบบเรียลไทม์ โดยใช้เทคนิค DNS-base และ Mining-base เป็นวิธีหนึ่งที่ใช้หลักการวิเคราะห์พฤติกรรมและการวิเคราะห์จากสภาพความคับคั่งของการจราจรในการเรียกใช้ DNS การติดตาม IP Address ที่เป็นต้นกำเนิดในการส่งแสปมเพื่อค้นหา Botnet โดยจะกระทำการวิเคราะห์แบบเรียลไทม์ซึ่งความสามารถแบบเรียลไทม์นี้จะช่วยในการโต้ตอบที่รวดเร็วแต่การวิเคราะห์ในลักษณะนี้จะไม่สามารถตรวจสอบโปรโตคอลและโครงสร้างที่เป็นอิสระได้ จึงได้นำเทคนิค Mining-base มาใช้ตรวจสอบเพื่อให้สามารถตรวจสอบในส่วนของ ตรวจสอบ Bot ที่ไม่รู้จัก ตรวจสอบโปรโตคอลและโครงสร้างที่เป็นอิสระ ตรวจสอบ Bot ที่เข้ารหัส

Keywords; Botnet,Botnet Detection,Malware,Spam, Phishing

I. บทนำ

บอตเน็ต[1] (BOTNET) หรือ roBOT NETwork เป็นภัยคุกคามต่อผู้ใช้งานอินเทอร์เน็ตรูปแบบใหม่ ซึ่งแฮ็กเกอร์เขียนโปรแกรม Botnet โดยใช้เทคนิคการโจมตีเครือข่ายอินเทอร์เน็ตด้วยโปรแกรมประสงค์ร้าย มัลแวร์ (Malware) ที่ซับซ้อนและมีรูปแบบที่หลากหลายกว่าไวรัสคอมพิวเตอร์หรือหนอนอินเทอร์เน็ตทั่วไป Botnet ที่ถูกสร้างขึ้นนี้อาจเป็นเครื่องมือที่ใช้ส่งสแปมเมล์ (Spam Mail) และ ฟิชชิ่ง (Phishing) การขโมยข้อมูลส่วนตัว ซึ่งเป็นวิธีการสร้างความเสียหายให้กับระบบเครือข่ายอินเทอร์เน็ต ลักษณะที่สำคัญของ Botnet ก็คือจะมีศูนย์กลางควบคุมและสั่งการโดยแฮ็กเกอร์อยู่ที่ใดที่หนึ่งบนอินเทอร์เน็ต กลไกการทำงานของ Botnet ถูกออกแบบให้มีการแพร่กระจายตัวเพื่อหาเครื่องใหม่ให้เข้ามาอยู่ในกลุ่มและมีความสามารถในการแก้ไขโปรแกรมของ Bot ที่ฝังตัวอยู่บนเครื่องคอมพิวเตอร์ผีดิบเพื่อเปลี่ยนแปลงรูปแบบการบุกรุก ลักลอบใช้งานและสั่งการผ่านศูนย์ควบคุม ซึ่งองค์ประกอบหลักของ Botnet ได้แก่เครื่องคอมพิวเตอร์สั่งการระยะ ไกลของแฮ็กเกอร์ เครื่องเซิร์ฟเวอร์ของห้องสนทนา Internet Relay Chat (IRC) ที่เป็นจุดนัดพบระหว่างกลุ่มของ Bot และแฮ็กเกอร์เพื่อรอรับคำสั่ง กลุ่มของ DNS เซิร์ฟเวอร์ซึ่งเป็นทางผ่านเพื่อทำให้ Bot สามารถหาเครื่องเซิร์ฟเวอร์ของห้องสนทนา IRC เจอได้ นอกจากนี้ยังมีกลุ่มของเครื่องคอมพิวเตอร์ต่าง ๆ บนเครือข่ายอินเทอร์เน็ตที่เป็นเป้าหมายของ Botnet และกลุ่มที่ได้กลายเป็นส่วนหนึ่งของ Botnet ไปแล้ว กระบวนการทำงานของ Botnet มีขั้นตอนดังรูปที่ 1



รูปที่ 1 แสดงขั้นตอนการทำงานของ Botnet

นักวิจัยได้เสนอวิธีการไม่กี่วิธี [2,3,4,5,6,7,8,9, 10,11,12,13,14] ในการตรวจสอบ Botnet วิธีการตรวจสอบเกือบทั้งหมดถูกออกแบบมาสำหรับการตรวจสอบ Botnet ใช้ IRC หรือ HTTP ตัวอย่างเช่น Snort [2] ถูกออกแบบมาเพื่อตรวจสอบ

ลายเซ็น IRC Botnet ที่รู้จักกันในรูปแบบชื่อเล่นของเครือข่ายการจัดกลุ่มและจำแนกตาม IRC อีกระบบ คือ BotSniffer [5] ถูกออกแบบสำหรับการตรวจสอบกิจกรรมของเครื่องควบคุมหลัก(C & C) กับเซิร์ฟเวอร์ส่วนกลาง (มีโปรโตคอล เช่น IRC และ HTTP1) อย่างไรก็ตาม Botnet มีการพัฒนาสามารถที่ค่อนข้างมีความยืดหยุ่น จะเห็นได้ว่าโปรโตคอลที่ใช้สำหรับเครื่องควบคุมหลัก มีวิวัฒนาการมาจาก IRC และ P2P นอกจากนี้ในช่วงชีวิตของ Botnet ยังสามารถเปลี่ยนเครื่องควบคุมหลัก กับที่อยู่ของเซิร์ฟเวอร์ส่วนกลางได้ ดังนั้นวิธีการตรวจสอบ Botnet ดังกล่าวข้างต้นตาม IRC หรือ HTTP อาจมีความผิดพลาด จึงมีการตรวจสอบแบบ [9] ซึ่งเป็นการวิเคราะห์พฤติกรรม โดยจะทำการวิเคราะห์จากสภาพความคับคั่งของการจราจรในการเรียกใช้ DNS จาก ผู้ให้บริการอินเตอร์เน็ต (ISP) และเครือข่ายขององค์กรที่ใช้งาน DNS ในการติดตาม IP Address ที่เป็นต้นกำเนิดในการส่งสแปมเพื่อค้นหาและระบุ Botnet โดยจะกระทำการวิเคราะห์ Botnet แบบเรียลไทม์ ซึ่งความสามารถแบบเรียลไทม์นี้จะช่วยในการโต้ตอบที่รวดเร็ว และบรรเทาผลกระทบที่จะเกิดขึ้น เช่น หาก IP Address ของโฮสต์ที่ตรวจพบว่าเป็น Botnet ซึ่งยังไม่ได้เปลี่ยนแปลง หรือเปลี่ยนแปลง IP Address ช้าพอที่จะให้การเคลื่อนไหวนั้นสามารถติดตามได้ในเวลาเรียลไทม์ ตัวดำเนินการของ DNS อาจจะตอบโต้ได้ทันท่วงที แต่การตรวจสอบรูปแบบนี้ยังไม่สามารถตรวจตอบ โปรโตคอลและโครงสร้างที่เป็นอิสระได้ จึงมีการตรวจสอบ Botnet ในลักษณะนี้ โดยใช้ Mining-based เช่น BotMiner [14] เป็นวิธีการตรวจสอบโดยอาศัยการบ่งชี้ความคับคั่งของ Botnet เนื่องจากการสื่อสารแบบ เครื่องควบคุมหลัก เป็นโปรโตคอลปกติที่ Botnet ใช้ ซึ่งการสื่อสารแบบนี้จะทำให้ความคับคั่งของระบบเป็นปกติ คือ ไม่เกิดสภาวะที่เครือข่ายใช้เวลาแฝงมากเกินปกติ (high latency network) ไม่เกิดสภาวะที่มีการคับคั่งของเครือข่ายมากเกินไป จึงทำให้วิเคราะห์ Botnet ใช้เวลามากและไม่เป็นแบบเรียลไทม์

ผู้วิจัยจึงคิดวิธีการตรวจสอบ Botnet แบบเรียลไทม์ โดยใช้เทคนิคของ DNS-based และ Mining based ผลที่ได้คือ การตรวจสอบ Botnet ที่เป็นโปรโตคอลและโครงสร้างที่เป็นอิสระในรูปแบบเรียลไทม์ โดยมีความผิดพลาดจากการตรวจสอบต่ำ

II. บทความและทฤษฎีที่เกี่ยวข้อง

การตรวจหา BOTNET วิธีการตรวจหา Botnet ปัจจุบันมีวิธีการค้นหาทั้งหมด 4 เทคนิคด้วยกันคือ

A. Signature-based Detection

เป็นระบบการตรวจหา botnet โดยอาศัยการตรวจหาลายเซ็นประจำตัวหรือพฤติกรรมประจำตัวของ botnet นั้นๆ จึงเป็นที่มาของอีกชื่อหนึ่งนั่นคือ signature based หรือ knowledge based detection ความรู้ในเรื่องของลายเซ็นประจำตัวหรือพฤติกรรมประจำตัวของ botnet มีประโยชน์ในการตรวจหา botnet คือ หากเราทราบถึงพฤติกรรมประจำตัวหรือกลไกการทำงานของ botnet เราจะสามารถตรวจจับได้ว่ามีการรัน botnet นั้นๆอยู่

ตัวอย่างเช่น Snort[2] ซึ่งเป็น open source ระบบตรวจจับการบุกรุก (IDS) ซึ่งตรวจสอบ network traffic เพื่อหาสัญญาณของการบุกรุก Snort มีการกำหนดค่าของกฎหรือลายเซ็นเพื่อล็อค traffic ที่ถือว่าน่าสงสัย[2] อย่างไรก็ตามเทคนิคการตรวจหาแบบ signature-based detection สามารถใช้สำหรับการตรวจหา Botnet ที่รู้จักเท่านั้น ดังนั้นการแก้ปัญหานี้ยังไม่เป็นประโยชน์สำหรับ Bot ที่ไม่รู้จัก

B. Anomaly-based Detection

การทำงานของระบบนี้จะเป็นการตรวจสอบ pattern ของข้อมูลหรือจะเรียกว่าพฤติกรรมต่างของ ข้อมูลที่วิ่งอยู่ในระบบเพื่อเรียน รู้ว่าอะไรคือสิ่งปกติและผิดปกติภายในระบบโดยที่ระบบจะมีกระบวนการเรียนรู้ ด้วยตัวเอง เหมือนดังเช่นกับระบบ spam filter โดยปกติแล้วระบบนี้จะถูกตั้งค่าโดยผู้ดูแลระบบเครือข่ายโดยที่ผู้ดูแล อาจจะกำหนดเส้นแบ่งว่าพฤติกรรมไหนถือว่าเป็นพฤติกรรมที่ปกติโดยอาจจะพิจารณา จาก traffic, พฤติกรรม, protocol หรือขนาดของข้อมูลเป็นต้น ดังนั้นจะทราบได้ทันทีว่าพฤติกรรมไหนเป็นพฤติกรรมที่เข้าข่ายการโจมตีระบบนั่นเองเนื่องจากระบบ Anomaly Based นั้นสามารถที่จะเรียนรู้ได้ด้วยตนเอง ดังนั้นระบบดังกล่าวนี้ก็จะสามารถเรียนรู้วิธีหรือพฤติกรรมใหม่ๆ ที่ใช้ในการโจมตีระบบได้นั่นเองแต่ก็อาจจะทำงานผิดพลาดได้นั่นหมายถึงไม่มี การส่งสัญญาณเตือนเมื่อมีการโจมตีเพราะเข้าใจว่าเป็นพฤติกรรมที่ปกติในระบบ เครือข่าย ตัวอย่างเช่น Botsniffer [3]

C. DNS-based Detection

เป็นการตรวจสอบหา botnet โดยอาศัยข้อมูลจาก DNS ที่สร้างโดย botnet นั้น เป็นวิธีการที่คล้ายกับ Anomaly-based Detection (การตรวจจับโดยอาศัยการตรวจความผิดปกติของ pattern ของข้อมูล) โดยจะเน้นที่การตรวจสอบความผิดปกติของสภาวะที่มีการคับคั่งของ DNS มากเกินไป คือเนื่องจากกลไกการทำงานของ bot เริ่มต้นการเชื่อมต่อโดยการเชื่อมต่อกับ C&C Server เพื่อรับเอาคำสั่ง เพื่อที่จะเข้าถึงกับ C&C Server bot จะต้องแสดง DNS queries เพื่อหาตำแหน่งที่ตั้งของ C&C Server ดังนั้นจึงสามารถตรวจหาความผิดปกติของสภาวะที่มีการคับคั่งของ DNS มากเกินไป โดยการเฝ้าสังเกตการคับคั่งของ DNS

ตัวอย่างเช่น Dagon [4] และ Kristoff [5] ใช้กลไกในการตรวจหา botnet โดยตรวจสอบโดเมนเนมที่มีมากกว่าปกติ หรือโดเมนเนมที่มี query rate บน DDNS มากเกินไป อย่างไรก็ตาม กลไกนี้ยังมีจุดอ่อนคือ หากใช้ DNS ปลอมจะไม่สามารถตรวจสอบได้

ในปี 2007, Choi et al, [15] นำเสนอวิธี anomaly-based โดยการตรวจสอบกิจกรรมกลุ่ม Botnet ในการจราจรใน DNS ซึ่งรูปแบบกิจกรรมกลุ่มใน DNS มีการส่งคำสั่งพร้อมกันโดย Bot กระจาย พวกเขามีการกำหนดคุณลักษณะเฉพาะของการเข้าชม DNS เป็นกิจกรรมกลุ่มที่จะแยกแบบสอบถาม DNS botnet จากการสอบถาม DNS ถูกต้องตามกฎหมาย เนื่องจากการจราจร DNS จะปรากฏในหลายขั้นตอนของ Botnet วงจรชีวิตก็เป็นไปได้ในการตรวจสอบบ็อตเน็ตโดยใช้คุณสมบัติกิจกรรมกลุ่มของการเข้าชม DNS Botnet พวกเขายังได้พัฒนากลไกที่ช่วยให้การตรวจสอบการโยกย้ายเซิร์ฟเวอร์ C & C นี้ วิธี anomaly-based มีประสิทธิภาพมากขึ้นกว่าวิธีก่อนหน้านี้และสามารถตรวจสอบ botnet ไม่คำนึงถึงชนิดของ bot และ botnet โดยดูที่กิจกรรมกลุ่มของพวกเขาในการจราจร DNS นอกจากนี้ยังสามารถตรวจสอบ botnets ที่มีการเข้ารหัสเนื่องจากมันจะใช้ข้อมูลที่ส่วนหัวของ IP อย่างไรก็ตามข้อเสียของวิธีนี้คือการใช้เวลาการประมวลผลสูงสำหรับการตรวจสอบเครือข่ายขนาดใหญ่

D. Mining-based Detection

เป็นวิธีการตรวจสอบโดยอาศัยการบ่งชี้ความคับคั่งของ botnet C&C เป็นการตรวจจับที่ค่อนข้างยาก แต่มีประสิทธิภาพสูง เนื่องจากการสื่อสารที่ C&C ใช้ เป็นโปรโตคอลปกติ ซึ่งการสื่อสารแบบนี้จะทำให้ความคับคั่งของระบบเป็นปกติ คือ ไม่เกิดสภาวะที่เครือข่ายใช้เวลาแฝงมากเกินปกติ (high latency network), จึงทำให้วิธี Anomaly-based Detection ไม่สามารถตรวจจับความคับคั่งของ C&C ได้

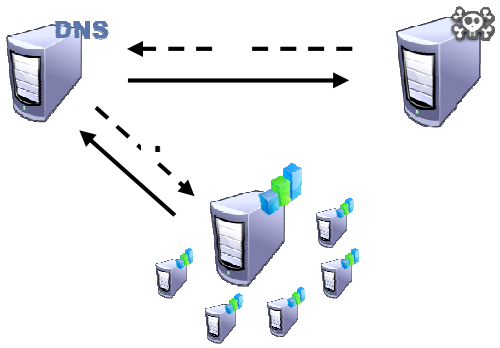
ตัวอย่างเช่น Rishi [6] คิดค้นโดย Geobl and Holz เป็นการเฝ้าระวังตรวจหาความผิดปกติของ IRC, nicknames, IRC servers, และ server ports ที่ไม่ปกติ ซึ่ง Rishi นี้มีจุดอ่อนคือจะไม่สามารถตรวจหา botnet ประเภท non-IRC ได้

ตารางที่ 1 เปรียบเทียบเทคนิคการตรวจสอบ Botnet ด้วยเทคนิคแบบต่างๆ

วิธีการตรวจสอบ		ตรวจสอบ Bot ที่ไม่รู้จัก	โปรโตคอลและโครงสร้างที่เป็นอิสระ	ตรวจสอบ Bot ที่เข้ารหัส	ตรวจสอบแบบเรียลไทม์	ความผิดพลาดจากการตรวจสอบต่ำ
Signature-based	[2]	✗	✗	✗	✗	✗
Anomaly-based	[3]	✓	✗	✗	✗	✗
	[4]	✓	✗	✓	✗	✓
	[5]	✓	✗	✓	✗	✓
DNS-based	[6]	✓	✗	✓	✗	✗
	[7]	✓	✗	✓	✗	✗
	[8]	✓	✗	✓	✗	✓
	[9]	✓	✗	✓	✓	✗
	[10]	✓	✓	✓	✗	✓
Mining-based	[11]	✓	✗	✗	✗	✗
	[12]	✓	✗	✗	✗	✗
	[13]	✓	✓	✓	✗	✓
	[14]	✓	✓	✓	✗	✓

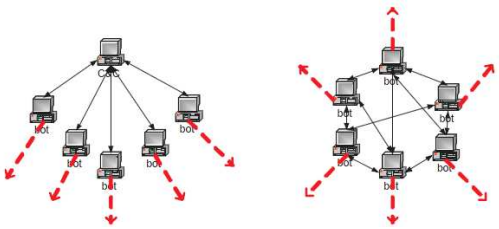
III. การทดลอง

อธิบายขั้นตอนการตรวจสอบ Botnet แบบเรียลไทม์ โดยใช้ DNS-base และ Mining-base ดังรูปที่ 2



รูปที่ 2 แสดงขั้นตอนการตรวจสอบ Botnet แบบเรียลไทม์ โดยใช้ DNS-base และ Mining-base

a. ตรวจสอบ IP Address ที่เป็นต้นกำเนิดในการส่งสแปม จะตรวจสอบจากสแปมที่ได้รับแล้วทำการส่ง IP Address ของเครื่องที่ส่งสแปม ไปยังเครื่องวิเคราะห์บอตเน็ตโดยลักษณะของบอตเน็ตจะมี 2 แบบคือแบบเครื่องควบคุมหลัก และ แบบ Peer-to-Peer ดังรูปที่ 3

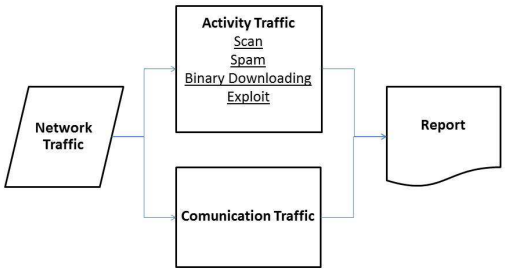


รูปที่ 3 แสดงบอตเน็ต แบบเครื่องควบคุมหลัก และแบบ Peer-to-Peer

อัลกอริทึ่มในการตรวจจับ Botnet โดยใช้ DNS

```
1   FOR k = 1 to n
2       IF (A_t1 => DN_k) is equal to (A_t  => DN_k)
3           similarity(A_t1 => IPList_k, A_t2 => IPList_k)
4           S = computed similarity
5       IF S > a , a  = Similarity threshold
6           DN_k is botnet domain name
7       ELSE IF S = - 1 THEN insert(BL, DN_k)  BL = blacklist
8       ELSE insert(W , DN_k)
9   ENDIF
End of Detect-BotDNS-Query
```

b. วิเคราะห์ IP Address จากฐานข้อมูลรายงาน Botnet และจัดเก็บ Network traffic ทำการนำ IP Address ของเครื่องที่ส่งสแปมมาตรวจสอบกับฐานข้อมูลรายงาน Botnet ที่ระบบวิเคราะห์ ทำการบันทึกไว้ และจัดเก็บ Network traffic สำหรับนำมาทำการวิเคราะห์ดังรูปที่ 4 โดยการวิเคราะห์จะทำในสองส่วนพร้อมๆกันคือ การวิเคราะห์ Activity Traffic และ Comunication Traffic คู่ขนานกันไป โดย Activity Traffic นั้นจะทำการตรวจจับการบุกรุกโดยใช้โปรแกรมโอเพนซอร์ส Snort เพื่อตรวจจับพฤติกรรมที่เข้าข่ายความผิดปรกติของการใช้งานเครือข่ายทั้งหมด คือการสแกน การสแปม การดาวน์โหลดไบนารี่ และการโจมตีช่องโหว่ในเครือข่าย ในขณะเดียวกันส่วนของ Comunication Traffic นั้นระบบจะจับกระแสข้อมูลในเครือข่าย ว่ามีการสื่อสารแบบใดบ้าง โดยอาศัยคุณสมบัติการบันทึกกระแสเครือข่ายที่เราท์เตอร์ส่วนใหญ่สามารถทำได้อยู่แล้ว โดยจะจำกัดความสนใจไปที่ TCP และ UDP และบันทึกข้อมูลต่อไปนี้ คือ เวลา, ระยะเวลา, IP ต้นทาง, พอร์ตต้นทาง, IP ปลายทาง, พอร์ตปลายทางและหมายเลขของแพ็คเก็ตและขนาดของข้อมูลในทั้งสองทิศทาง และส่งผลการตรวจสอบกลับไปยังเครื่อง DNS-base



รูปที่ 4 แสดงถึงกระบวนการวิเคราะห์ Botnet โดยใช้เทคนิค Mining-base

V. สรุปและงานที่ต้องทำต่อ

ผู้วิจัยได้ทำการศึกษา Botnet แบบ RaelTime โดยใช้ DNS-based และ Mining based จากผลการทดลองที่ผ่านมา ทำให้สรุปผลได้ว่าการประยุกต์ใช้เทคนิคในการตรวจจับร่วมกันระหว่าง DNS-based และ Mining based ทำให้ความถูกต้องในการตรวจจับนั้นไม่ได้ลดลงแต่อย่างใด แต่กลับมีผลดีคือสามารถตรวจจับ Botnet แบบ RaelTime ได้เพิ่มขึ้น ซึ่งถือว่าเพิ่มประสิทธิภาพของการตรวจจับได้เป็นอย่างดี เราวางแผนว่าจะพัฒนาฐานข้อมูลในส่วน ที่ตรวจจับให้มีข้อมูลเยอะกว่านี้ เพื่อให้

a

b

ลดความผิดพลาดจากการตรวจสอบต่ำลงกว่าเดิม และเพิ่มความรวดเร็วในการตรวจจับให้มากขึ้น โดยพัฒนาระบบให้มีขนาดใหญ่ขึ้นด้วย

เอกสารอ้างอิง

[1] ดร.กมล เขมะรังษี และ กิติศักดิ์ จิรวรรณกูล, “บอตเน็ต ภัยรูปแบบใหม่บนอินเทอร์เน็ต,” 10 สิงหาคม 2548.

[2] Snort IDS web page. http://www.snort.org, March 2006.

[3] J.R. Binkley and S.Singh, “An algorithm for anomaly-based botnet detection,” in Proc. USENIX Steps to Reducing Unwanted Traffic on the Internet Workshop (SRUTI’06), , 2006, pp 43–48.

[4] A. Karasaridis, B. Rexroad, and D. Hoeflin, “Wide-scale botnet detection and characterization,” in Proc. 1st Workshop on Hot Topics in Understanding Botnets, 2007.

[5] G. Gu, J. Zhang, and W. Lee, “Botsniffer: Detecting botnet command and control channels in network traffic,” in Proc. 15th Annual Network and distributed System Security Symposium (NDSS’08), 2008.

[6] D. Dagon, “Botnet Detection and Response, The Network is the Infection,” in OARC Workshop, 2005.

[7] J. Kristoff, “Botnets,” in 32nd Meeting of the North American Network Operators Group, 2004.

[8] A. Schonewille and D.J. van Helmond. “The Domain Name Service as an IDS,” Master’s Project, University of Amsterdam, Netherlands, Feb 2006, http://staff.science.uva.nl/~delaat/snb-2005-2006/p12/report.pdf

[9] N. F. A. Ramachandran and D. Dagon, “Revealing botnet membership using dnsbl counter-intelligence,” in Proc. 2nd Workshop on Steps to Reducing Unwanted Traffic on the Internet (SRUTI ’06), 2006.

[10] H. Choi, H. Lee, H. Lee, and H. Kim, “Botnet Detection by Monitoring Group Activities in DNS Traffic,” in Proc. 7th IEEE International Conference on Computer and Information Technology (CIT 2007), 2007, pp.715-720.

[11] J. Goebel and T. Holz, “Rishi: Identify bot contaminated hosts by irc nickname evaluation,” in Proc. 1st Workshop on Hot Topics in Understanding Botnets, 2007.

[12] W. Strayer, D. Lapsley, B. Walsh, and C. Livadas, Botnet Detection Based on Network Behavior, ser. Advances in Information Security. Springer, 2008, PP. 1-24.

[13] M. M. Masud, T. Al-khateeb, L. Khan, B. Thuraisingham, K. W. Hamlen, “ Flow-based identification of botnet traffic by mining multiple log file,” in Proc. International Conference on Distributed Frameworks & Applications (DFMA), Penang, Malaysia, 2008.

[14] G. Gu, R. Perdisci, J. Zhang, and W. Lee, “Botminer: Clustering analysis of network traffic for protocol- and structure independent botnet detection,” in Proc. 17th USENIX Security Symposium, 2008