## ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА КОМП'ЮТЕРНА ТЕХНІКА

УДК 004.492.3

**О. С. Савенко**, канд. техн. наук, доц.;
**С. М. Лисенко**, канд. техн. наук, доц.;
**А. Ф. Крищук**

# МОДЕЛЬ БОТНЕТ-МЕРЕЖ

*Запропоновано узагальнену модель функціонування ботнет-мережі. Запропоновано модель бота, що функціонує в межах ботнет-мережі. Отримані моделі враховують життєвий цикл ботнет-мережі та її властивості, що проявляються в процесі функціонування і дозволяють здійснювати пошук нових ботів на конкретних комп'ютерних системах.*

### Вступ

Ботом є мережне програмне забезпечення, яке може взаємодіяти з системами або користувачами. Ботнет — це мережа приватних комп'ютерних систем, інфікованих шкідливим програмним забезпеченням, що контролюється без відома користувача, наприклад, для здійснення розсилки спаму або інших зловмисних дій. Походить від скорочень англійських слів «robot» — робот і «network» — мережа [1].

На етапі експлуатації комп'ютерних мереж досить часто відбуваються зміни в їх конфігурації, використовуваному програмному і апаратному забезпеченні. Тому необхідно постійно здійснювати діагностування мережі на наявність ботів, що може міститись в комп'ютерних системах, які входять до її складу.

Комп'ютерна система, яка включає в себе бот, ймовірно була атакована іншим зловмисним програмним забезпеченням для інфікування ботом. Зловмисник може обманом користувача встановити програмне забезпечення разом з ботом (користувач відкриває вкладення електронної пошти, де встановлений бот). Зловмисник може використати уразливість на використовуваній машині, щоб дозволити віддалене виконання коду і встановити «бекдор» [2].

До основних напрямків використання ботнет-мереж можна віднести розподілені атаки на відмову в обслуговуванні (DDoS-атаки), збір конфіденційної інформації, розповсюдження реклами і небажаних повідомлень, а також поширення ботів та іншого шкідливого програмного забезпечення. Згідно з дослідженнями «Лабораторії Касперського», Україна є джерелом 12 % DDoS-трафіка у світі [3].

Використання ботнет-мереж може порушити в конкретній комп'ютерній системі цілісність, достовірність та конфіденційність інформації. Тому задача діагностування комп'ютерних мереж на наявність елементів ботнет-мереж є актуальною [2].

Дослідження поведінки і будови ботнет-мереж дозволяє запропонувати нові їх моделі з метою виявлення і зменшення зловмисного впливу на комп'ютерні системи.

### Постановка задачі

Щоб створити ефективні методи виявлення ботнет-мереж, необхідно побудувати узагальнену модель функціонування ботнет-мережі, а також бота, що функціонує в межах цієї ботнет-мережі. Модель ботнет-мережі має враховувати її життєвий цикл, особливості та властивості в процесі функціонування. Також необхідно довести врахування в запропонованих моделях усіх особливостей функціонування ботнет-мереж шляхом порівняння реального зловмисного програмного забезпечення із запропонованими моделями.

Об'єктом дослідження є процес функціонування ботнет-мережі та її ботів як складових.

Предметом дослідження є модель бота в ботнет-мережі.

### Модель ботнет-мережі

Розглянемо модель ботнет-мережі $A$ як систему керування ботами. Система отримує від об'єкта

керування події, що характеризують зміни його стану, і вона сама запитує поточні параметри об'єкта, що також вважається вхідним впливом на систему $A$.

Представимо модель ботнет-мережі множиною [4, 5]

$$A = \langle T, Q, K \rangle, \tag{1}$$

де $T$ — тип топології; $Q$ — складові мережі; $K$ — комунікація в мережі.

Тип топології задамо таким чином:

$$T = \langle S, M, I, R, H \rangle, \tag{2}$$

де елементи вказаної множини відображають такі топології: $S$ — топологія зірка, де боти під'єднуються безпосередньо до командного центру, як показано на рис. 1а.

Ця топологія забезпечує швидку реакцію ботнет-мережі на команди і водночас має такі недоліки:

— існує центральна точка відмови;

— одному вузлу відомо IP-адреси всіх ботів мережі; вузол містить всі дані про ботнет мережу;

— масштабованість: існує обмеження на кількість підключених ботів до одного командного сервера, які він може ефективно обробляти.

$M$ — мультисерверна, де сервери ретельно скоординовані один з одним для створення видимості наявності централізованої системи. Сервери спілкуються двонапрямлено один з одним, і таким чином забезпечують можливість керування усіма ботами без наявності єдиного центрального вузла. Якщо один із серверів став недоступним, то боти, які виявили неполадку, сповіщають підлеглим ботам про це і, відповідно, можуть переключитися на інший сервер (рис. 1б).

$I$ — ієрархічна, яка може містити в собі або командний сервер, або набір серверів, і лише визначені вузли під'єднані до них. Такий підхід не сприяє надійності в разі відмови центрального вузла, але зменшує кількість ботів, які мають бути інформовані про розташування центрального серверу (С) (рис. 1в).

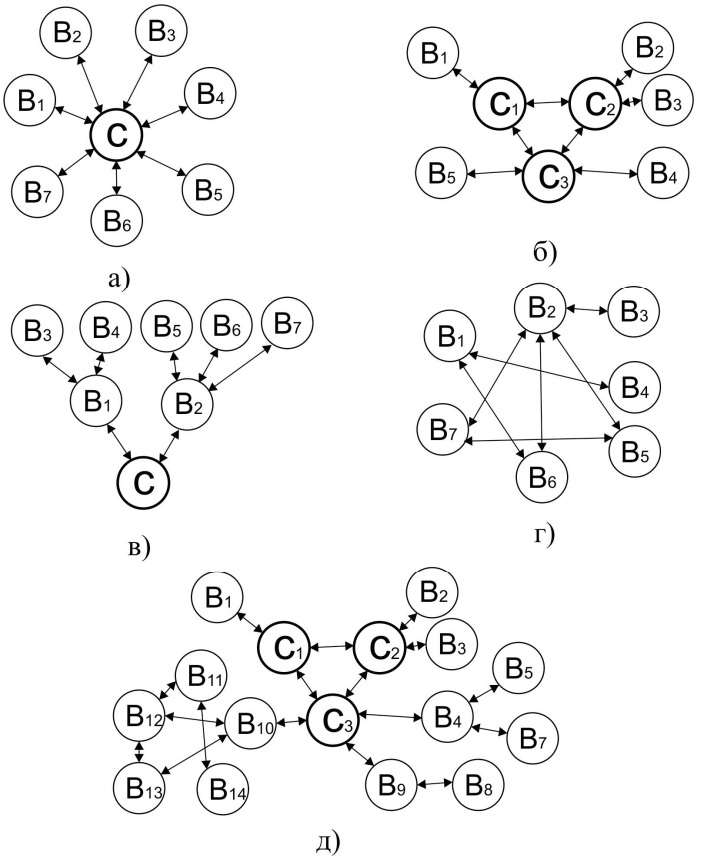


Рис. 1. Топології ботнет мереж: а — зірка; б — мультисерверна; в — ієрархічна; г — випадкова; д — гібридна

$R$ — випадкова, де вузли з'єднані між собою випадковим чином. Кожен вузол може виступати командним центром. Така топологія має високий рівень відмовостійкості, оскільки демонтаж одного вузла має незначний вплив на інші вузли ботнет-мережі. З іншого боку, збільшується затримка виконання команд, тому що для досягнення конкретного вузла можуть знадобитися кілька переходів (рис. 1г).

$H$ — гібридна, де вузли з'єднуються з командним центром як безпосередньо, так і ієрархічно відносно центрального вузла або випадковим чином. Поєднання з'єднань кількох видів топологій дозволяє підвищити рівень відмовостійкості і реакцію ботів на подані команди (рис. 1д).

Складовими мережі $Q = \langle P, E \rangle$ є такі елементи:

$P$ — вузли вірусної мережі, де вузлами виступають інфіковані КС, боти яких є складовою ботнет-мережі, а також сервери командних центрів, якщо такі присутні в ботнет-мережі визначеної топології; $E$ — зв'язок між вузлами.

Нехай вузли ботнет-мережі $P = \langle B, C \rangle$ включають в себе такі елементи:

$B = \{b_1, b_2, ..., b_n\}$ — множина ботів, що належить ботнет-мережі; $C = \{c_1, c_2, ..., c_n\}$ — множина командних центрів ботнет-мережі.

Командний центр має контролювати встановлену кількість ботів, призначених для здійснення зловмисних дій. Командний центр керується людиною-оператором, яка контролює боти, що перебувають під контролем. Відоме розташування конкретної ботнет-мережі або командного центра не означає, що зловмисник знаходиться поблизу [6].

У разі топології мережі «зірка» боти керуються одним командно-контролюючим центром, $C = \{c_1\}$; для топології мультисерверної мережі боти керуються визначеною кількістю командних центрів, $C = \{c_1, c_2, ..., c_n\}$; для випадкової топології мережі кожен бот є командним центром, а тому $B = C$.

Приймемо комунікацію в мережі між ботами та командно-контролюючими центрами як

$$K = \langle L, O \rangle, \tag{3}$$

де $L = \{l_1, l_2, ..., l_n\}$ — множина вузлів, перелік яких містить визначений вузол $(L \in (B \cup C))$;

$O = \cup p_i^{pr}$ — трафік, де $p_i$ — пакети трафіка; *pr* — протокол передачі даних в мережі.

Введемо множину $X$, $X \in O$ як множину запитів командних центрів до бота, причому $X = X_j \cup X_V$, де $X_j$ — множина запитів, які відповідають за технічне обслуговування бота, $X_j = \{X_{j1}, X_{j2}, ..., X_{jn}\}$; $X_{j1}$ — автентифікація пароля; $X_{j2}$ — запит оновлення модуля; $X_{j3}$ — запит оновлення бота; $X_{j4}$ — запит оновлення списку можливих підключень; $X_{j5}$ — запит виведення конфігурації бота; $X_{j6}$ — запит перезапуску бота; $X_{j7}$ — запит статусу бота; $X_{j8}$ — запит виведення інформації про користувача; $X_{j9}$ — запит відображення поточних підключень до бота; $X_{j10}$ — запит відображення ботів в ботнет-мережі; $X_{j11}$ — запит зміни пароля користувачу; $X_{j12}$ — запит зміни власного пароля; $X_{j13}$ — запит знищення бота; $X_V$ — множина запитів, які відповідають за атакуючі дії ботнет-мережі, $X_V = \{X_{V1}, X_{V2}, ..., X_{Vn}\}$ (рис. 2); $X_{V1}$ — початок атаки на відмову ресурсу; $X_{V2}$ — переведення бота в режим «проксі-сервера»; $X_{V3}$ — виконання «фішинг»-атак; $X_{V4}$ — виконання «клікфорд»-атак; $X_{V5}$ — виконання «спам»-атак; $X_{V6}$ — виконання сканування активних КС; $X_{V7}$ — виконання визначення ОС визначеної КС; $X_{V8}$ — виконання сканування портів визначеної КС; $X_{V9}$ — виконання завантаження файлів на визначену КС; $X_{V10}$ — виконання дій поширення бота.

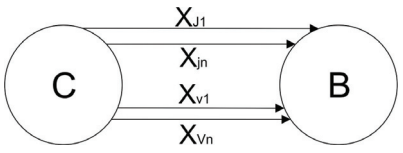


Рис. 2. Передача команд від командного центру до бота

Нехай $U$ — множина відповідей бота *b* командному центру, $U \in O$ $U = U_J \cup U_V$, де $U_J$ — множина запитів, які відповідають за технічне обслуговування бота $U_J = \{u_{J0}, u_{J1}, ..., u_{Jn}\}$; $U_V$ — множина запитів, які відповідають за звітність бота про виконані дії; $U_V = \{u_{V1}, ..., u_{Vn}\}$. $u_{J0}$ — запит, який підтверджує під'єднання бота до ботнет-мережі (рис. 3).

Кожний бот може отримати команду в будь-який момент часу, якщо $(d_i \in D) \cup (L \neq 0)$.

Кожен вузол мережі (чи то бот, чи то КЦ) надсилає команди тільки вузлам з множини *L*.

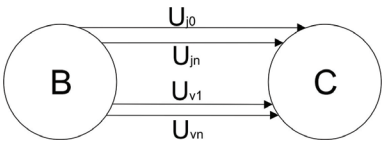


Рис. 3. Передача команд від бота до командного центру

### Модель бота

Приймемо узагальнену модель бота як множину

$$b = \langle G, Z, Y, W, w_0, \sum, \alpha \rangle, \tag{4}$$

де $G_b = G_J \cup G_V$ — множина всіх подій, на які реагує бот *b*, причому $G_J = \{g_{J1}, g_{J2}, ..., g_{Jn}\}$ —

зовнішні впливи на бот, якими є команди від командного центру $X_J \in G_J$; $G_V = \{g_{V1}, g_{V2}, ..., g_{Vn}\}$ — внутрішні впливи на бот, якими є події всередині КС; $g_{V1}$ — робота антивірусних засобів; $g_{V2}$ — блокування або закриття портів; $g_{V3}$ — робота контролю облікових записів ОС; $g_{V4}$ — видалення користувачем частин бота.

Введемо $Z = Z_J \cup Z_V$ як множину всіх впливів бота *b* на комп'ютерну систему, де розміщується бот *b*. Розглядаються впливи двох видів. Перший вид — вплив безпосередньо на КС у разі отримання відповідної команди *x*, $Z_J = \{z_{J1}, z_{J2}, ..., z_{Jn}\}$; $z_{j1}$ — відслідковування натиснень клавіш клавіатури; $z_{j2}$ — збереження зображення екрана; $z_{j3}$ — збереження даних з веб-камери; $z_{j4}$ — пошук електронних адрес на ІКС; $z_{j5}$ — збереження конфіденційних ключів, номерів платіжних карток і паролів; $z_{j6}$ — збереження списків користувачів електронних ресурсів; $z_{j7}$ — збереження списків користувачів електронних ресурсів; $z_{j8}$ — пошук ресурсів на ІКС; $z_{j9}$ — відправлення файлів на вказані ресурси; $z_{j10}$ — генерація трафіку для атаки на відмову; $z_{j11}$ — завантаження модулів бота; $z_{j12}$ — деактивація антивірусних засобів; $z_{j13}$ — зміна пароля користувача (КС або програми обміну повідомленнями); $z_{j14}$ — видалення інших ботів; $z_{j15}$ — знищення бота.

Другий вид впливів — вплив на КС в процесі виконання дій з множини *Y*, $Z_V = \{z_{V1}, z_{V2}, ..., z_{Vn}\}$; $z_{V1}$ — призупинення служб і процесів; $z_{V2}$ — запуск необхідних служб і процесів; $z_{V3}$ — зміна конфігураційних файлів; $z_{V4}$ — відкриття системних портів; $z_{V5}$ — закриття системних портів;

Позначимо $Y_a = \{y_1, y_2, ..., y_n\}$ як множину впливів бота на атакований ресурс; $y_1$ — генерування запитів до визначеного ресурсу встановленими ботами; $y_2$ — надсилання визначеним адресатам електронні листи; $y_3$ — надсилання визначеним адресатам миттєві повідомлення через соціальні мережі; $y_4$ — надсилання визначеним адресатам повідомлення через менеджери миттєвих повідомлень і програми IP-телефонії; $y_5$ — сканування активних КС; $y_6$ — визначення ОС визначеної КС; $y_7$ — сканування портів визначеної КС; $y_8$ — переведення КС в режим «проксі-сервера».

Введемо $W = \{w_0, ..., w_9\}$ як скінченну множину станів бота для отримання необхідних прав згідно з політикою доступу, щоб керувати КС і реалізовувати поставлені задачі, де $W_i (i = \overline{0,9})$ [7]:
$w_0$ — початковий стан бота, розповсюдження; $w_1$ — визначення функціонуючої КС; $w_2$ — визначення типу операційної системи; $w_3$ — сканування системних портів; $w_4$ — визначення сервісів; $w_5$ — інфікування КС; $w_6$ — деактивація антивірусних засобів; $w_7$ — приховування слідів проникнення; $w_8$ — створення прихованих можливостей переміщення; $w_9$ — реалізація загроз.

Перед інфікуванням КС бот розміщується в програмах, які могли б їх поширити. Цими носіями виступають троянські програми і «worm»-віруси. Методами поширення виступають «спам»-повідомлення в менеджерах обміну повідомленнями (icq, skype, irc, і т. д.), соціальних мережах (Vkontakte, Facebook, Twiter, Мой Мир, LinkedIn і т. д.), електронній пошті, а також різного роду носії інформації та електронні ресурси.

«Worm»-віруси використовують помилки в операційних системах, створених розробниками, або іншим шкідливим програмним забезпеченням. Спочатку визначається, чи КС є активною в цей час (наприклад, виконання команди «ping» на вказаний ресурс). Далі проводиться пошук КС з встановленою операційною системою, на помилки якої спрямоване це ШПЗ. Далі скануються системні порти, через які відбуватиметься проникнення. Після інфікування очікується активація дій бота, за якою відбувається деактивація антивірусних засобів, перевірка на наявність віртуальних машин та інших ботів у КС. Потім проводиться автентифікація бота в ботнет-мережі і за необхідності відкриваються додаткові порти для завантаження необхідних модулів в залежності від призначення відповідного бота.

Подальші дії бота зосереджені на забезпеченні конфіденційності перебування в КС і очікуванні

команд для виконання закладених зловмисних дій.

Розглянемо приклад фунціонування бота ботнет-мережі KaiBot.a, який для комунікації зберігає список відомих ботів фіксованого розміру. Бот підтримує зв'язок із іншими відомими ботами, що забезпечує зловмиснику можливість керування ботнет-мережею. Відповідно до закладених властивостей бот періодично обмінюється поточними списками з іншими доступними вузлами. Це дозволяє боту дізнатися про інші інфіковані КС у ботнет-мережі та вилучити неактивні боти зі списку. У роботі [6] досліджено, що боти KaiBot. a містять список з 200 вузлів. Коли бот дізнається про інші існуючі вузли у структурі, його список збільшується до максимально можливих 500 записів. Кожен запис містить IP-адресу вузла і час останньої активності. Якщо ж кількість відомих «колег» перевищує 500, бот залишає у списку лише 500 найактивніших за встановлений проміжок часу. Під час обміну списками боти копіюють 99 записів від свого партнера, додають свою IP-адресу та поточний час, і відправляють його випадково обраному партнеру. Основною метою такої ботнет-мережі є викрадення конфіденційної інформації.

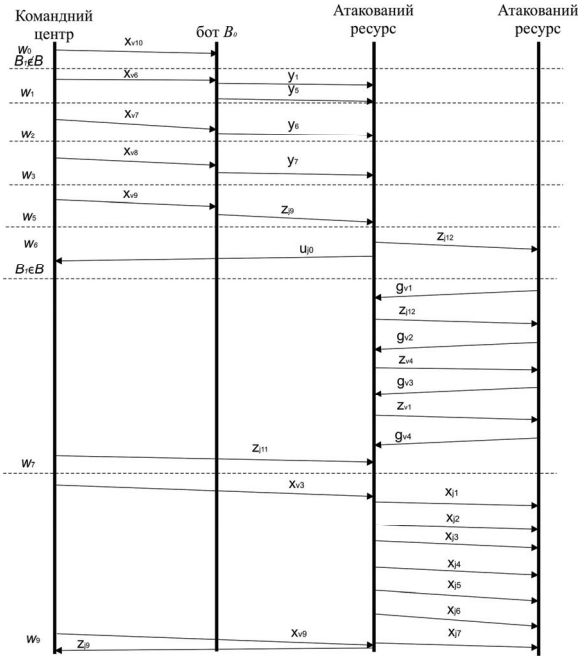


Рис. 4. Схема виконання команд ботнет-мережі

Зобразимо узагальнений процес виконання запитів цієї ботнет-мережі схемою, показаною на рис. 4.

Отже, маємо вузли ботнет-мережі $P_b = \langle B, C \rangle$, в якій наявна множина ботів $B = \{b_1, b_2, ..., b_i\}$, $i = \overline{1,200}$, та один командний центр ботнет-мережі $C = \{c_1\}$.

Бот $B = \{b_1\}$ генерує множину запитів, які відповідають за обслуговування бота $U = U_J \cup U_V$.

Також командний центр генерує множину запитів, які відповідають за атакуючі дії ботнет-мережі, які відбуваються в момент часу, визначений зловмисником, $X_V = \{X_{V3}, X_{V5}, X_{V6}, X_{V7}, X_{V8}, X_{V9}\}$, та множину запитів технічного обслуговування $X_j = \{X_{j1}, X_{j2}, X_{j3}, X_{j4}, X_{j12}, X_{j13}\}$; $X_{V3}$ — виконання «фішинг»-атак; $X_{V5}$ — виконання «спам»-атак; $X_{V6}$ — виконання сканування активних КС; $X_{V7}$ — виконання визначення ОС визначеної КС; $X_{V8}$ — виконання сканування портів визначеної КС; $X_{V9}$ — виконання завантаження файлів на визначену КС; $X_{V10}$ — виконання процесу поширення бота.

Бот мережі здійснює множину впливів бота на атакований ресурс $Y_a = \{y_2, y_3, y_7, y_9\}$.

Проаналізуємо життєвий цикл (ЖЦ) такої ботнет-мережі. Для цього розглянемо стани бота в різні моменти часу його функціоування. В момент розповсюдження початковий стан бота $\forall B = \{b_1, b_2, ..., b_i\}$, $i = \overline{1,200}$ $W = \{w_0\}$, $X = \{X_{V10}\}$; командний центр цієї ботнет-мережі генерує множину запитів до бота $X = \{X_{V6}\}$, проводиться перевірка активних КС $Y = \{y_1, y_5\}$, $W\{w_1\}$, та визначення типу операційної системи $X = \{X_{V7}\}$, $Y = \{y_6\}$, $W = \{w_2\}$. Далі проводиться сканування портів на виявлених активних КС, тобто $X = \{X_{V8}\}$, $Y = \{y_7\}$, $W = \{w_3\}$, $X \in O$. Під час здійснення запиту $X = \{X_{V9}\}$ визначеній КС завантажується тіло бота $Z = \{z_{j9}\}$, $W = \{w_5\}$.

На наступному етапі ЖЦ $W = \{w_6\}$ виконується вимкнення існуючих антивірусних засобів $Z = \{z_{j12}\}$ і під'єднання бота до ботнет-мережі виконанням запиту, $U = \{u_{j0}\}$, $B_1 \in B$.

Далі відбувається призупинення модулів або повної деактивації антивірусного програмного забезпечення, якщо це програмне забезпечення було відновлене $G = \{g_{v1}\}$, $Z = \{z_{j12}\}$; у разі блокування або закриття портів бот реагує відкриттям необхідних системних портів $G = \{g_{v2}\}$,

$Z = \{z_{v4}\}$; під час активного контролю облікових записів відбувається вимкнення відповідних служб і процесів $G = \{g_{v3}\}$, $Z = \{z_{v1}\}$; і, відповідно, якщо відбулось видалення частини бота, проводиться їхнє завантаження, $G = \{g_{v4}\}$, $Z = \{z_{j11}\}$, $W = \{w_7\}$.

На наступному етапі ЖЦ виконується очікування запитів $X$ від командного центру шляхом прослуховування встановлених портів, $W = \{w_9\}$. Бот, після отримання запиту виконання «фішингової» атаки, $X = \{X_{V3}\}$, здійснює атаку $Z = \{Z_{j1}, Z_{j2}, Z_{j3}, Z_{j4}, Z_{j5}, Z_{j6}, Z_{j7}\}$; відслідковує натискання клавіш, операції з електронними грошовими рахунками, програми автентифікації в онлайн-іграх та інтернет аукціонах і т. д.

Після запиту командного центру $X = \{X_{V9}\}$ відбувається вивантаження усіх збережених файлів з бота на командний центр, $Z = \{Z_{j9}\}$.

Наступним етапом є повторення стану ЖЦ $W = \{w_0, w_1, w_2, w_3, w_5\}$ для подальшого поширення ботів у мережі, де командний центр цієї ботнет-мережі генерує множину запитів до КС, тобто $X = \{X_{V6}, X_{V7}, X_{V8}\}$, $X \in O$. Такий вплив активує множину дій бота $Y = \{y_5, y_6, y_7\}$, $G_V = \{g_{V4}\}$. Під час здійснення команди $X = \{X_{V9}\}$ визначеній КС завантажується тіло бота $Z = \{z_{j9}\}$.

## Висновок

В результаті проведеного дослідження запропоновано узагальнену модель функціонування ботнет-мережі. Запропоновано модель бота, що функціонує в межах ботнет-мережі. Отримані моделі враховують життєвий цикл ботнет-мережі та усі її особливості та властивості в процесі функціонування.

Для перевірки вірності запропонованих моделей було доведено врахування в запропонованих поведінкових моделях усіх особливостей функціонування ботнет-мереж шляхом порівняння реального зловмисного програмного забезпечення із запропонованими моделями.

Запропоновані моделі в майбутньому дадуть можливість побудувати методи ефективного діагностування КС на наявність нового зловмисного програмного забезпечення, що ґрунтується на ботнет-мережах.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Oxford Dictionaries [Електронний ресурс]. — Режим доступу: http://oxforddictionaries.com.

2. Savenko O. Multi-agent Based Approach of Botnet Detection in Computer Systems / O. Savenko, S. Lysenko, A. Kryschuk // Computer Networks, Springer Berlin Heidelberg, 19th International Conference Szczyrk Poland, 2012. — P. 171—180.

3. LB.UA [Электронный ресурс] : Украина — один из главных источников DDoS-атак в мире. — Режим доступа: http://economics.lb.ua

4. Botnet Detection Based on Network Behavior / [D. Lapsely, Robert Walsh, Carl Livadas, W. Strayer Timothy]. — Springer Science+Business Media, LLC, Botnet Detection Countering the Largest Security Threat, 2008. — 165 p.

5. Cooke E. The zombie roundup : Understanding, detecting, and disrupting botnets / E. Cooke, F. Jahanian, D. McPherson // Wksh. Steps to Reducing Unwanted Traffic on the Internet, — 2005.

6. Borup L. Peer-to-peer botnets : A case study on Waledac / L. Borup. — Master's thesis, Technical University of Denmark, 2009.

7. Structured peer-to-peer overlay networks: Ideal botnets command and control infrastructures? / [C. R. Davis, S. Neville, J. M. Fernandez, J. M. Robert] // Euro. Symp. Research in Computer Security, 2008.

8. Porras P. A multi-perspective analysis of the Storm (Peacomm) worm / P. Porras, H. Saidi, V. Yegneswaran. — Tech. rep., Computer Science Laboratory, SRI International, 2007.

9. Measurements and mitigation of peer-to-peer-based botnets : A case study on Storm worm / [T. Holz, M. Steiner, F. Dahl, E. Biersack, F. Freiling] // USENIX Wksh. Large-Scale Exploits and Emergent Threats, 2008.

Рекомендована кафедрою захисту інформації

Стаття надійшла до редакції 2.01.2013
Рекомендована до друку 6.03.2013

***Савенко Олег Станіславович*** — доцент, ***Лисенко Сергій Миколайович*** — доцент, ***Крищук Андрій Федорович*** — аспірант.

Кафедра системного програмування, Хмельницький національний університет, Хмельницький