

ИССЛЕДОВАНИЕ МЕХАНИЗМОВ ЗАЩИТЫ ОТ АТАК DDOS: ИМИТАЦИЯ ПРОТИВОБОРСТВА ИНТЕЛЛЕКТУАЛЬНЫХ АГЕНТОВ В СЕТИ ИНТЕРНЕТ

И.В. Котенко, А.В. Уланов

**НИГ компьютерной безопасности
Санкт-Петербургский институт информатики и
автоматизации РАН**

«РусКрипто'2008», 3 - 6 апреля 2008 г.



План доклада

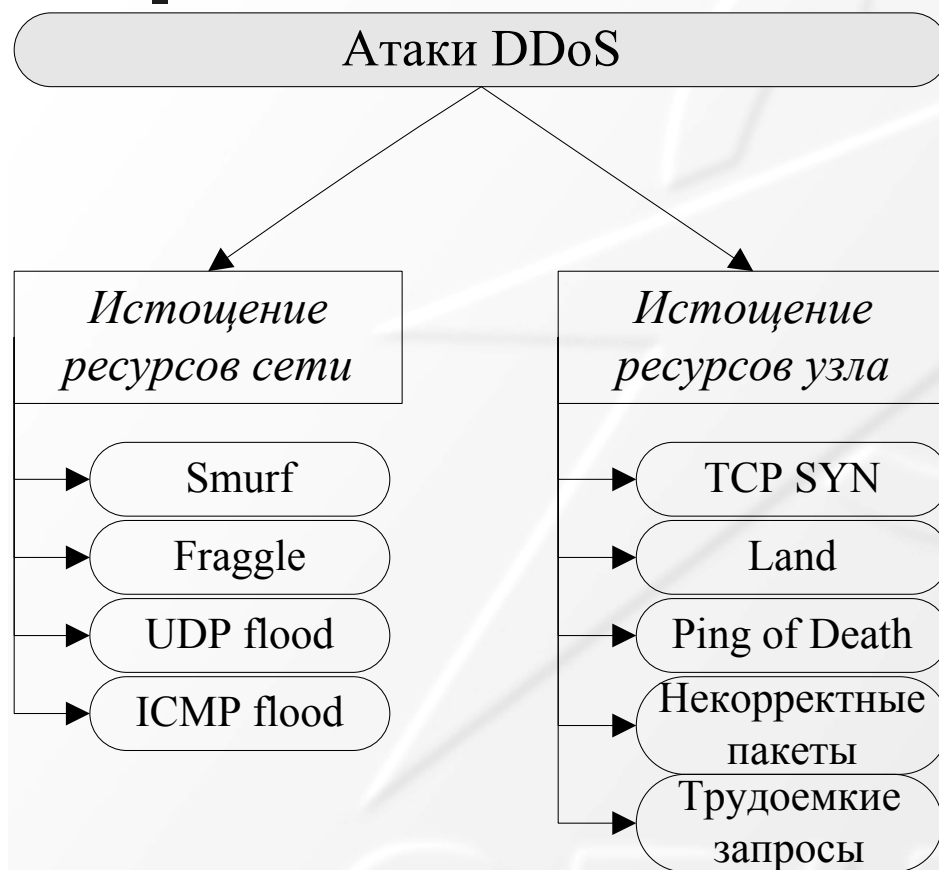
- Введение
- Атаки DDoS и механизмы защиты
- Подход к моделированию
- Среда моделирования
- Эксперименты
- Заключение



Обоснование проблемы

- **Цель работы:** разработка общего подхода и инструментальной среды для многоагентного моделирования противоборства команд программных агентов в компьютерных сетях (на примере DDoS-атак и механизмов защиты от них).
- **Теоретические задачи:** исследование новых механизмов защиты
- **Прикладные задачи:** оценка безопасности существующих сетей, рекомендации по построению перспективных систем защиты

Атаки DDoS



- DDoS – Distributed Denial of Service – распределенный отказ в обслуживании



- Реализация атак DDoS: выход из строя хостов, служб, сегментов сети

Механизмы защиты от DDoS атак

- Общий подход к защите от DDoS атак:
 - сбор информации о нормальном трафике
 - сравнение текущего трафика с модельным
 - прослеживание источников аномалий и выдача рекомендаций
 - применение выбранных контрмер






Кооперативные механизмы защиты от DDoS атак

- **Управление ресурсами**
 - Server Roaming
 - Market-based Service Quality Differentiation
 - Transport-aware IP router architecture
- **Аутентификация**
 - Transport-aware IP router architecture
 - Secure Overlay Services
- **Механизмы отслеживания (traceback)**
 - ACC pushback
 - COSSACK
 - Perimeter-based DDoS defense
 - DefCOM
 - Gateway-based

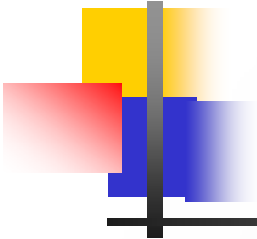
Многоагентное моделирование киберпротивоборства





Основные работы, используемые в качестве базиса для исследований

- **Агентно-ориентированное моделирование:** N.Jennings, M.Wooldridge и др.
- **Командная работа агентов:** P.Cohen (общие намерения), B.Grosz, S.Kraus (разделяемые планы); M.Tambe (комбинированные подходы) и др.
- **Системы вывода, основанные на предсказании намерений и планов оппонента:** E.Charniak (формулировка задачи распознавания как задачи абдуктивного вывода); H.Kautz, J.Allen (распознавание плана на основе идентификации минимального множества высокоуровневых действий, которые достаточны для объяснения наблюдаемых действий) и др.
- **рефлексивные процессы:** В.А.Лефевр, В.Е.Лепский, Д.А.Поспелов и др.
- **теоретико-игровое моделирование:** J.Nash, G.Zlotkin, J.Rosenschein, T.Sandholm, Ю.Б. Гермейер, А.И.Кондратьев и др.
- **Системы кооперативных распределенных грамматик** и грамматические модели агентских систем: M.ter Beek, J.Gaso, J.Kelemen, J.Dassow и др.
- **Моделирование атак на компьютерные сети:** [Ritchey *et al*-00], [Swiler *et al*-01], [Ortalo *et al*-01], [Sheyner *et al*-02] и др.
- **Моделирование процессов защиты информации**, в т.ч. моделей аутентификации и разграничения доступа, виртуальных частных сетей, инфраструктуры открытых ключей, обнаружения вторжений и др.



Работы в области многоагентного моделирования

- Классические подходы:
 - Теория общих намерений [Cohen 1991]
 - Теория общих планов [Grosz 1996]
 - Комбинированный подход, система Steam [Tambe 1997]
- Системы многоагентного моделирования
 - GRATE*: общая ответственность [Jennings 1995]
 - OAA: доска объявлений [Martin 1999]
 - CAST: общая ментальная модель [Yen 2003]
 - RETSINA-MAS: комбинации всевозможных ролей агентов [Giampara 2002]
 - Robocup soccer: ориентация на собственную модель мира [Stankevich 1999]
 - COGNET/BATON: взаимодействие людей и агентов [Zachary 1996]
 - Team-Soar: «многоуровневая теория» [Kang 2001]



Предлагаемый подход

- Кибернетическое противоборство представляется в виде взаимодействия различных команд программных агентов
- Процессы происходят в среде, задаваемой моделью Интернета
- Выделяются команды агентов атаки и защиты
- Знания о предметной области содержатся в онтологиях
- Агенты взаимодействуют по протоколам и в соответствии со сценариями
- Команды взаимодействуют между собой: противоборствуют, кооперируются, адаптируются



Основные положения подхода

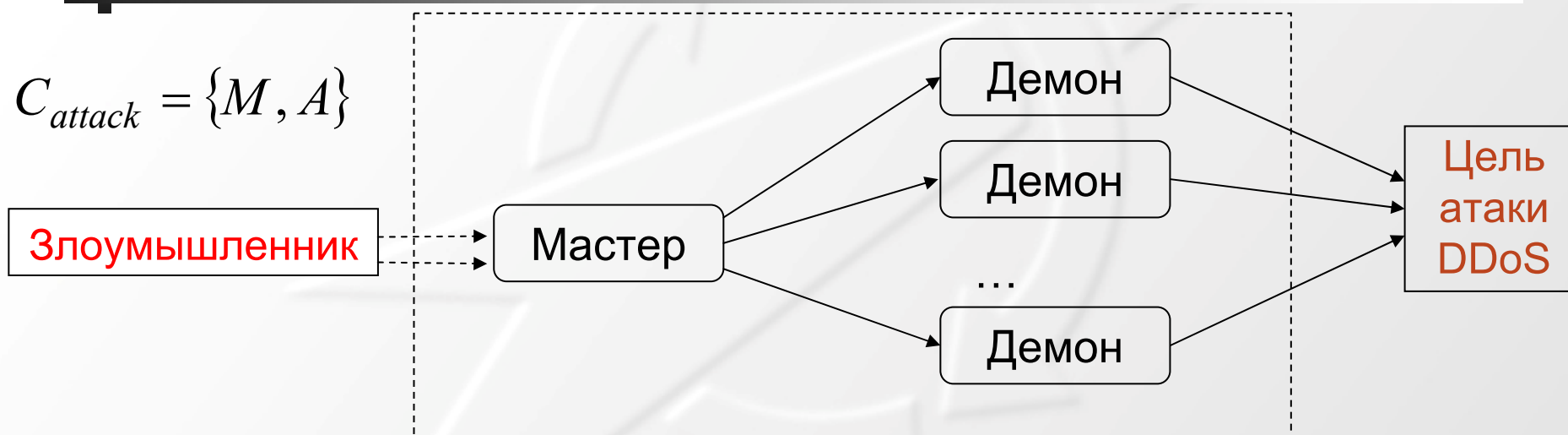
- Предлагаемый подход базируется на **комбинировании** элементов теории общих намерений, теории разделяемых планов и комбинированных подходов
- **Структура команды агентов** описывается в терминах иерархии групповых и индивидуальных ролей в различных сценариях действий
- **Спецификация иерархии планов** действий осуществляется для каждой из ролей. **Для каждого плана описываются:**
 - (1) начальные условия, когда план предлагается для исполнения;
 - (2) условия, при которых план прекращает исполняться;
 - (3) действия, выполняемые на уровне команды, как часть общего плана
- **Назначение ролей и распределение планов** между агентами выполняется в два этапа: (1) сначала план распределяется в терминах ролей, (2) каждой из ролей ставится в соответствие агент



Процедуры поддержки командной работы

- 1. Процедуры обеспечения согласованности действий агентов в команде** (*группе, индивидуально*) по некоторому общему плану.
- 2. Процедуры мониторинга и восстановления функциональности команды** (*группы, индивидуально*) за счет переназначения “утерянных” ролей тем членам команды, которые в состоянии выполнить эту работу
- 3. Процедуры обеспечения селективности коммуникаций**; основываются на расчете важности того или иного сообщения с учетом его “стоимости” и выгоды, получаемой при этом.

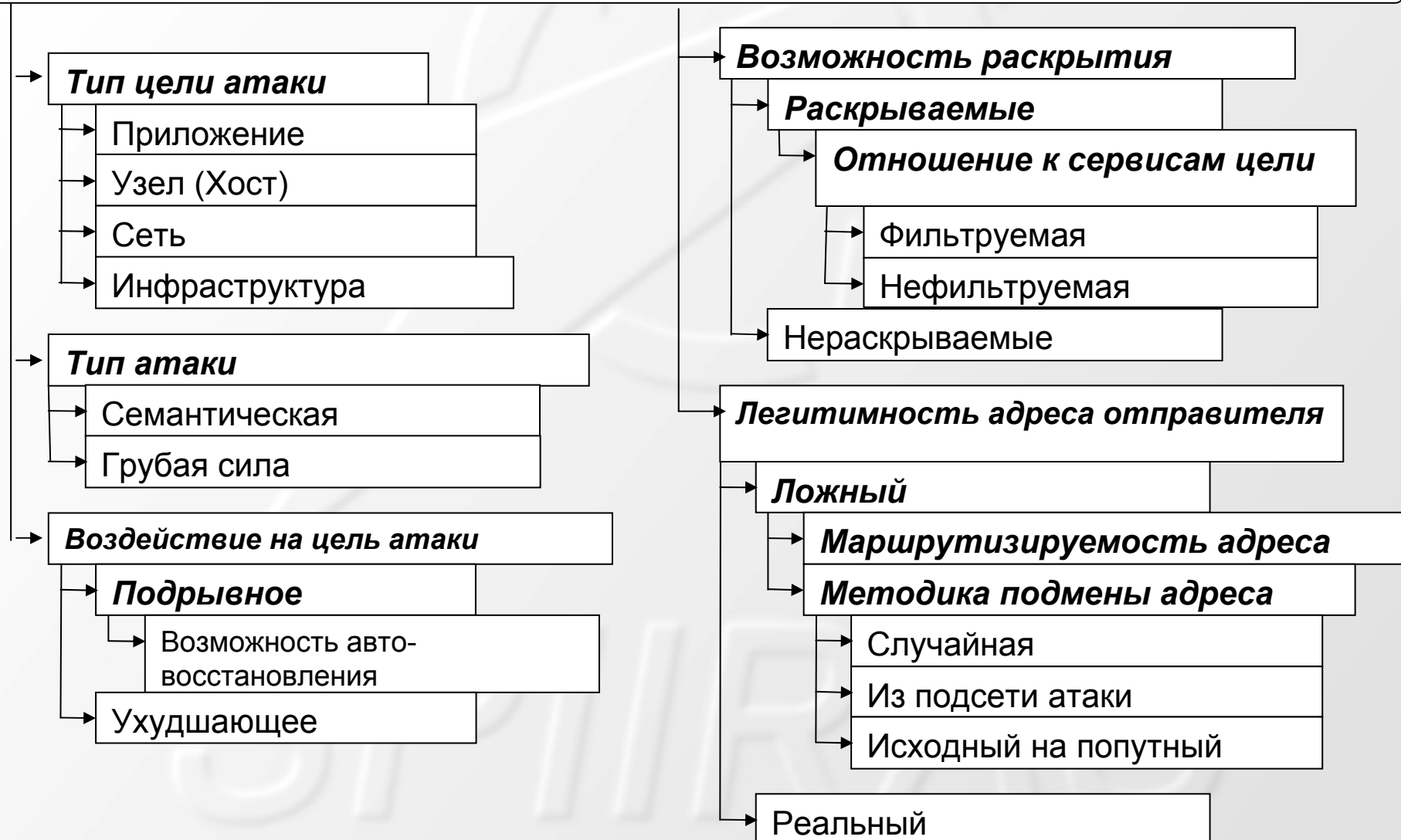
Команда атаки



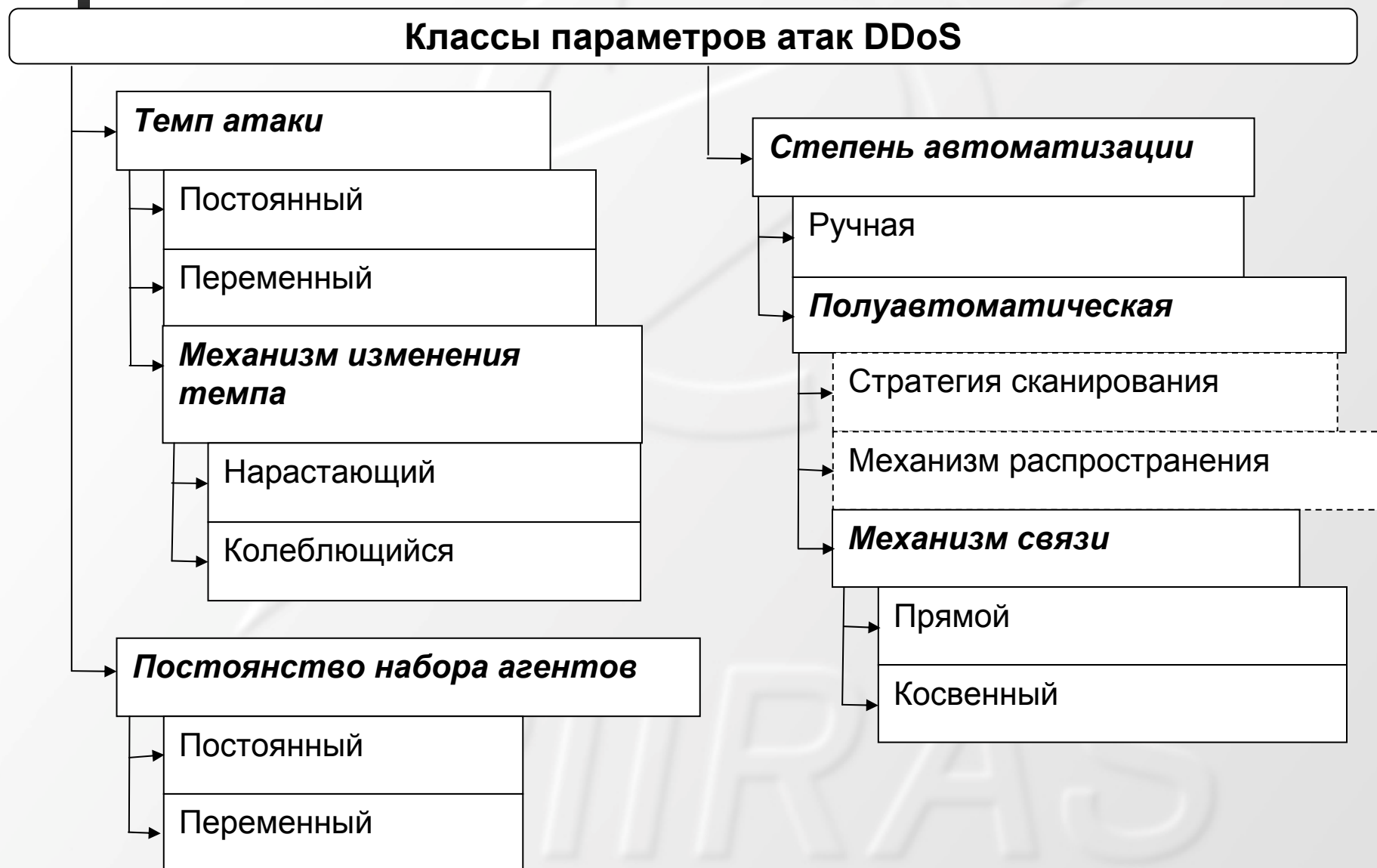
- **Атака DDoS:** глобальная цель достигается скоординированными усилиями многих компонентов
- **«Демон»** – исполнитель атаки
 - В начале работы посылает «мастеру» свой адрес и порт
- **«Мастер»** – координатор атаки
 - Составляет список работоспособных «демонов»
 - Получает команду атаки от злоумышленника
 - Посылает работоспособным «демонам» команду атаки: IP адрес и порт цели, интенсивность (пакетов в секунду)

Параметры моделирования атак (1)

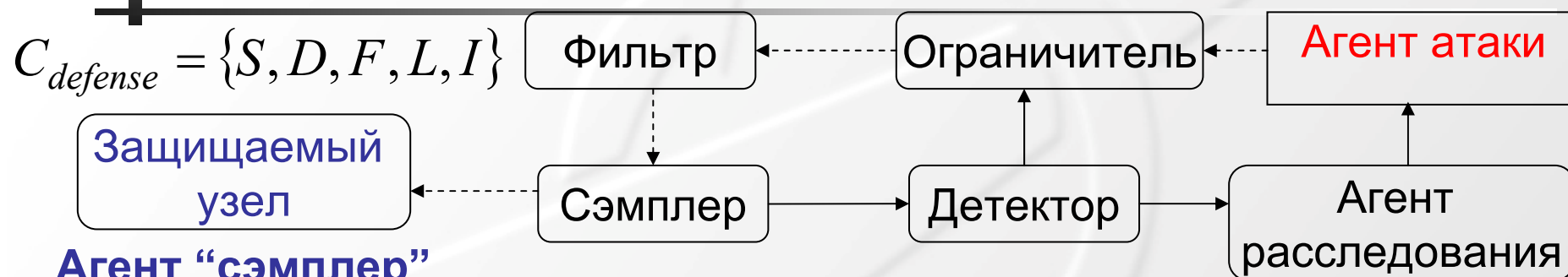
Классы параметров атак DDoS



Параметры моделирования атак (2)



Команда защиты



Агент «сэмплер»

- Сбор модельных данных для каждого узла по сетевым пакетам
- Выдача модельных данных на запрос «детектора»

Агент «детектор»

- Прием сообщения о работоспособности других агентов
- Запрос данных от «сэмплеров»
- Прием решения об атаке
- Посылка сообщения со списком подозрительных узлов «фильтру» и агенту «расследования», директиву ограничивать трафик

Агент «фильтр» – прием данных от «детектора» и фильтрация

Агент «расследования» – отслеживание источника атаки и его обезвреживание

Агент «ограничитель» – ограничение трафика


Параметры моделирования защиты (1)



Параметры моделирования защиты (2)

Классы параметров механизмов защиты от атак DDoS





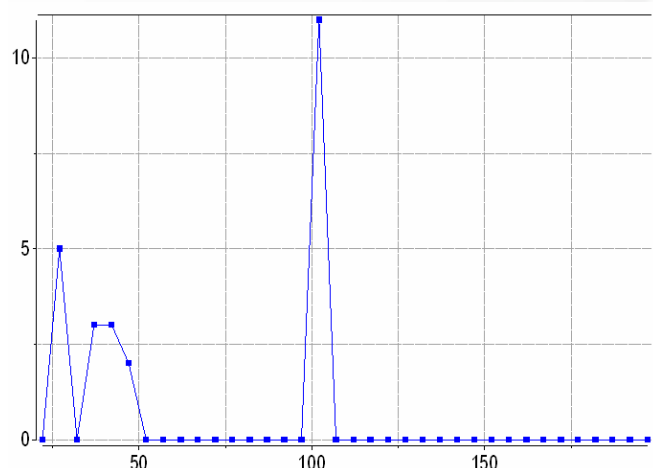
Примеры методов обнаружения вредоносного трафика

- **Hop counts Filtering (HCF):** заключается в формировании таблиц подсетей и количества скачков до них. Предполагается, что пакеты из одной и той же сети проходят от отправителя до получателя одинаковое количество хопов (скачков). Вначале составляется таблица, в которой узлы группируются по количеству хопов. При обнаружении атаки система, реализующая HCF, вычисляет количество хопов пришедшего пакета и сравнивает его с табличным значением.
- **Source IP address monitoring (SIPM):** используется предположение, что во время атаки появляется большое количество новых адресов клиентов. Вначале производится формирование базы IP-адресов “легитимных” клиентов. В реальном времени система собирает статистику по пакетам – количество новых для системы IP-адресов за заданные отрезки времени. Если эта величина остается в пределах нормы, то новые адреса заносятся в базу, если нет – осуществляется фильтрация.
- **Bit Per Second (BPS):** позволяет обнаружить атакующих по превышению порога нормального трафика. Вначале определяется «допустимый» порог для трафика на основе запросов легитимных клиентов.

Построение модели нормального трафика

■ Режим обучения:

- Легитимные клиенты обращаются к защищаемому серверу, а он обрабатывает их запросы, таким образом создавая выборку нормального трафика.
- Аналогичным образом создается выборка вредоносного трафика, при этом задействуются атакующие агенты команды атаки.



■ HCF

74 (std::vector<AR_NormHop *>) ...p[0].ad_statsnapp.*(nhpv.getV...

(std::vector<AR_NormHop *>) coop_methods.d_firewall.tcpApp[0].ad_statsnapp.*(nhpv.getV...

or<AR_NormIP *>) ...p[0].ad_statsnapp.*(nipv....

AR_NormIP *) coop_methods.d_firewall.tcpApp[0].ad_statsnapp.*(nipv...

v....

statsnapp.*(hsv...

> {

orPtr()[0] = IP=10.0.0.14	Time=23.1392
orPtr()[1] = IP=10.0.0.16	Time=22.002
orPtr()[2] = IP=10.0.0.64	Time=22.0022
orPtr()[3] = IP=10.0.0.65	Time=23.0002
orPtr()[4] = IP=10.0.0.66	Time=23.0001
orPtr()[5] = IP=10.0.0.12	Time=35.1574
orPtr()[6] = IP=10.0.0.13	Time=32.0373
orPtr()[7] = IP=10.0.0.20	Time=33.8253
orPtr()[8] = IP=10.0.0.15	Time=39.8271
orPtr()[9] = IP=10.0.0.17	Time=37.3458
AR_NormIP * (nipv.getVectorPtr())[10] = IP=10.0.0.18	Time=39.9285

ectorPtr()[8] = IP=10.0.0.17 Hop=7 Time=588.996

ectorPtr()[9] = IP=10.0.0.15 Hop=7 Time=582.381

ectorPtr()[10] = IP=10.0.0.18 Hop=6 Time=593.668

ectorPtr()[11] = IP=10.0.0.11 Hop=7 Time=587.323

ectorPtr()[12] = IP=10.0.0.19 Hop=7 Time=595.469

ectorPtr()[13] = IP=10.0.0.33 Hop=6 Time=580.118

ectorPtr()[14] = IP=10.0.0.34 Hop=5 Time=580.12

ectorPtr()[15] = IP=10.0.0.35 Hop=4 Time=580.121

ectorPtr()[16] = IP=10.0.0.36 Hop=5 Time=580.123

ectorPtr()[17] = IP=10.0.0.37 Hop=6 Time=580.125

ectorPtr()[18] = IP=10.0.0.23 Hop=5 Time=580.126

ectorPtr()[19] = IP=10.0.0.24 Hop=4 Time=580.127

ectorPtr()[20] = IP=10.0.0.25 Hop=3 Time=580.128

ectorPtr()[21] = IP=10.0.0.26 Hop=4 Time=580.129

■ SIPM

■ BPS

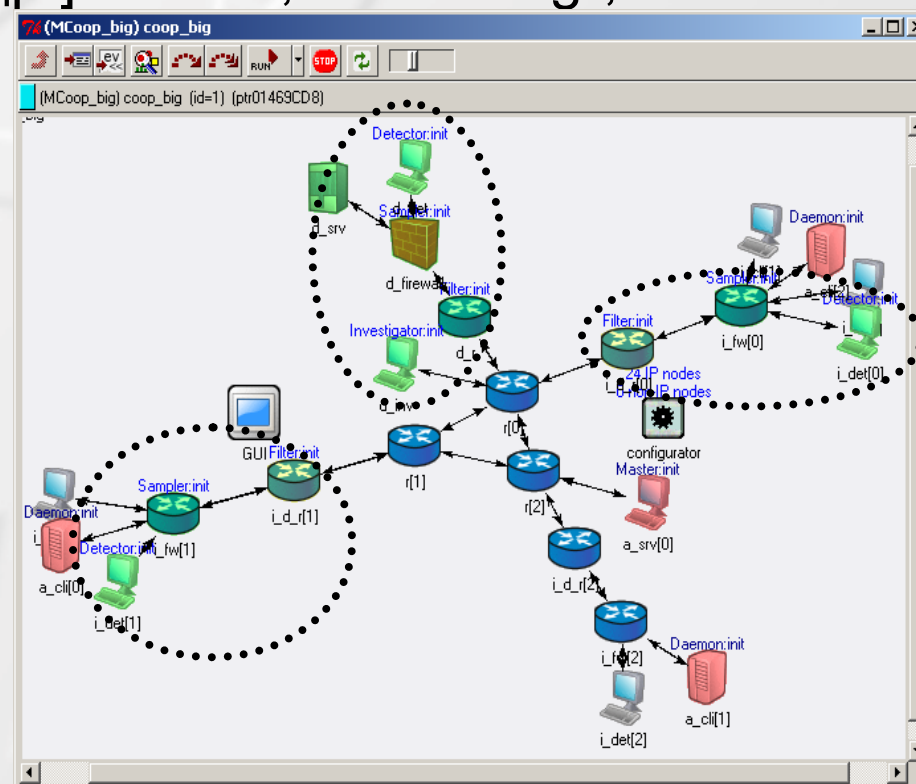
Кооперативные механизмы защиты

■ Модели кооперативного взаимодействия:

- DefCOM [Миркович и др.]: “Alert generator”, “Rate limiter”, “Classifier”
- COSSACK [Пападополус и др.]: “snort”, “watchdog”, filter

Предложенные:

- *без кооперации;*
- *кооперация на уровне фильтров;*
- *кооперация на уровне сэмплеров;*
- *слабая кооперация;*
- *полная кооперация.*



Архитектура среды моделирования





Среда моделирования

OMNeT++ Discrete Event Simulation System

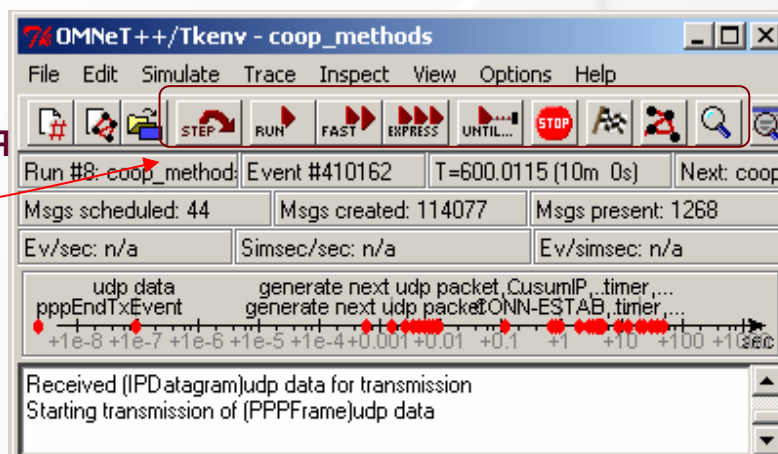
- Моделируемые процессы происходят в сети Internet
- Был проведен анализ пакетов моделирования: NS2, OMNeT++ INET Framework, SSF Net, J-Sim INET Framework и др.
- Выдвинутым требованиям удовлетворяет OMNET++ INET Framework (надстройка над OMNET++)
- OMNET++ – симулятор дискретных событий, которые происходят внутри модулей
- Модуль имеет шлюзы, через которые по каналам передаются сообщения
- INET Framework был доработан и дополнен новыми модулями для корректного моделирования механизмов атаки и защиты
- На его основе разрабатывается многоагентная среда моделирования борьбы за ресурсы в сети Internet

Окно

```

graph TD
    waiting --> ad_topapp
    ad_topapp --> a_samplerdrv
    a_samplerdrv --> ad_statsnapp

```



Моделируемая сеть

«РусКрипто'2008», 3 - 6 апреля 2008 г.

76(cOutVector) ...irewall.ppp[2].thru[0].thru...

(cOutVector) coop_methods.d_firewall.ppp[2].thru[0].thru

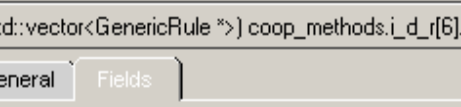
25423.2149

12942.226

461.237024

26.0115029 372.714907 600.011503

Last value: t=579.86242 (9m 39s) value=1028.64 Options...

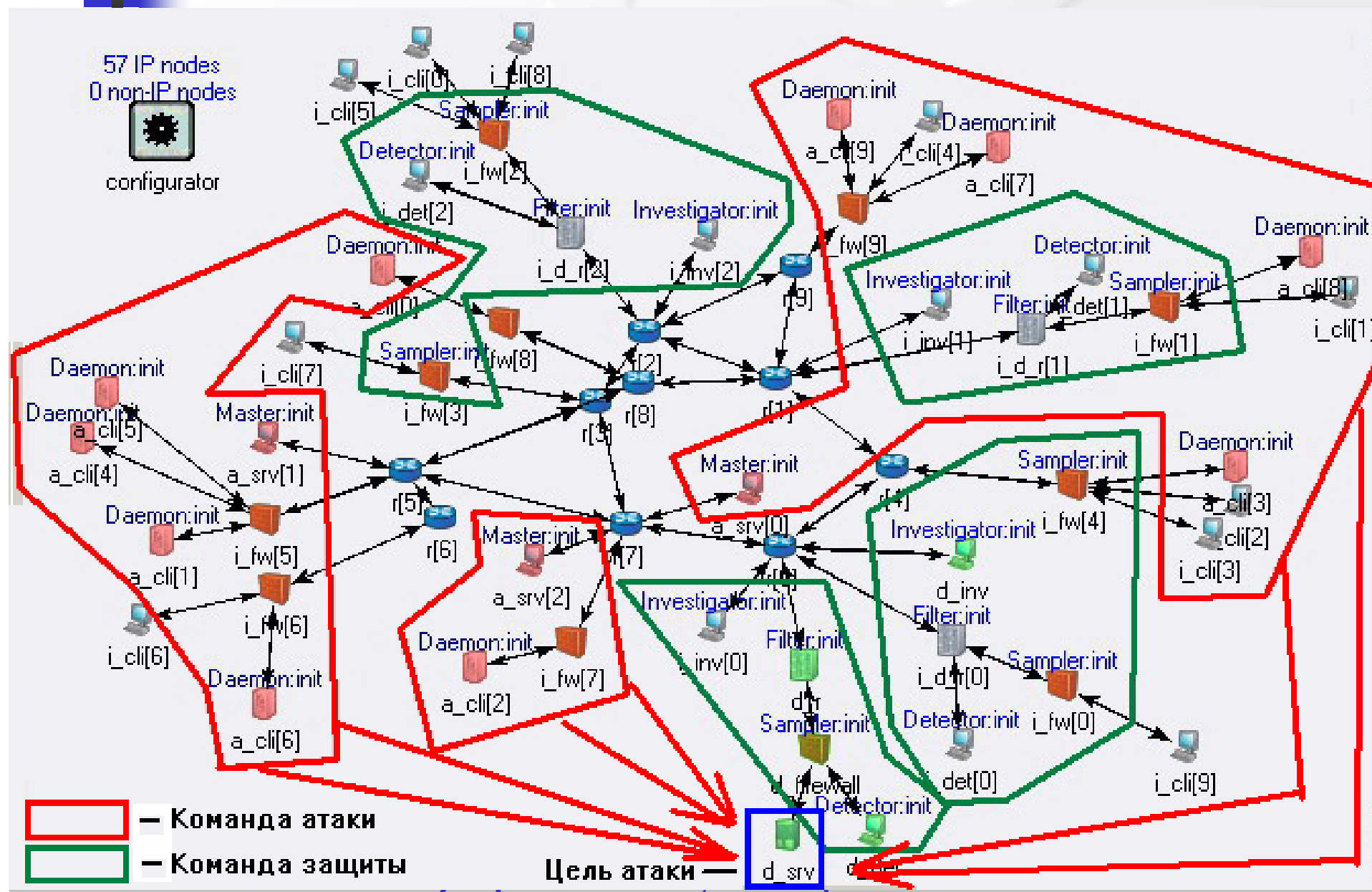


```
class std::vector<GenericRule *> {  
    GenericRule * rules_vector[0] = 10.0.0.61 1  
    GenericRule * rules_vector[1] = 192.168.0.1 1  
    GenericRule * rules_vector[2] = 192.168.0.10 1  
    GenericRule * rules_vector[3] = 192.168.0.11 1  
    GenericRule * rules_vector[4] = 192.168.0.12 1
```

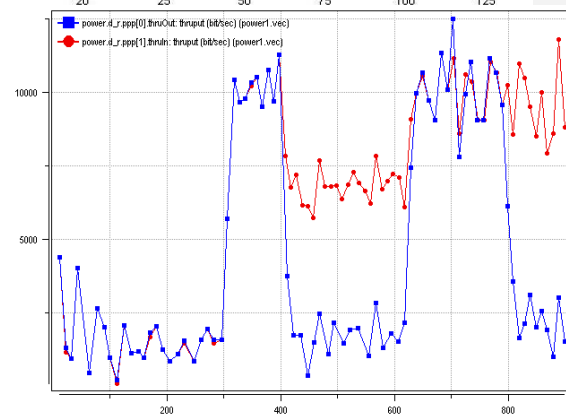
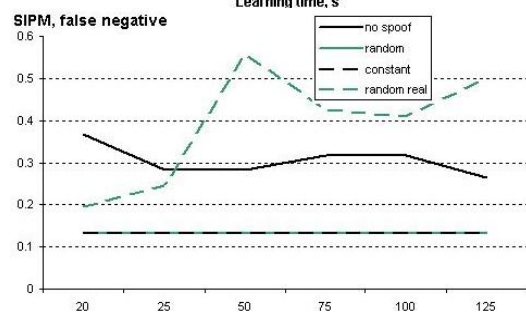
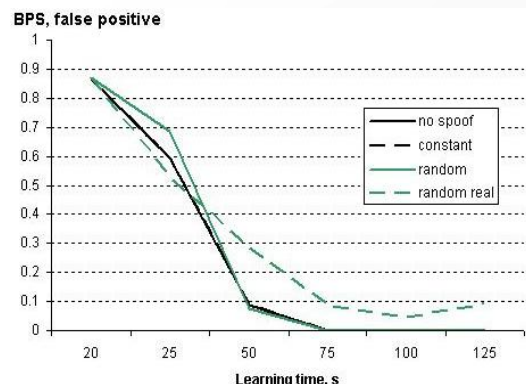
```

graph TD
    a((a)) --> a1((a1))
    a1 --> a2((a2))
    a1 --> a3((a3))
    a --- DRV[DRV]
    a1 --- ACTIONS[ATTACK TEAM ACTIONS]
    a2 --- EST[TEAM ESTABLISHING]
    a3 --- ATTACK[ATTACKING]
  
```

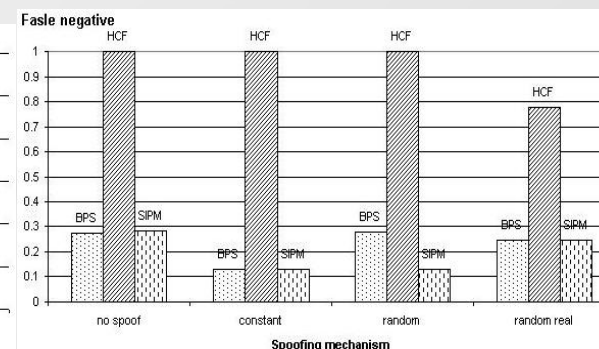
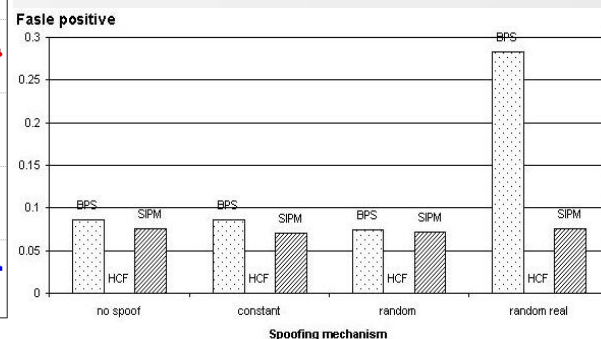
Пример моделируемой сети



Эксперименты

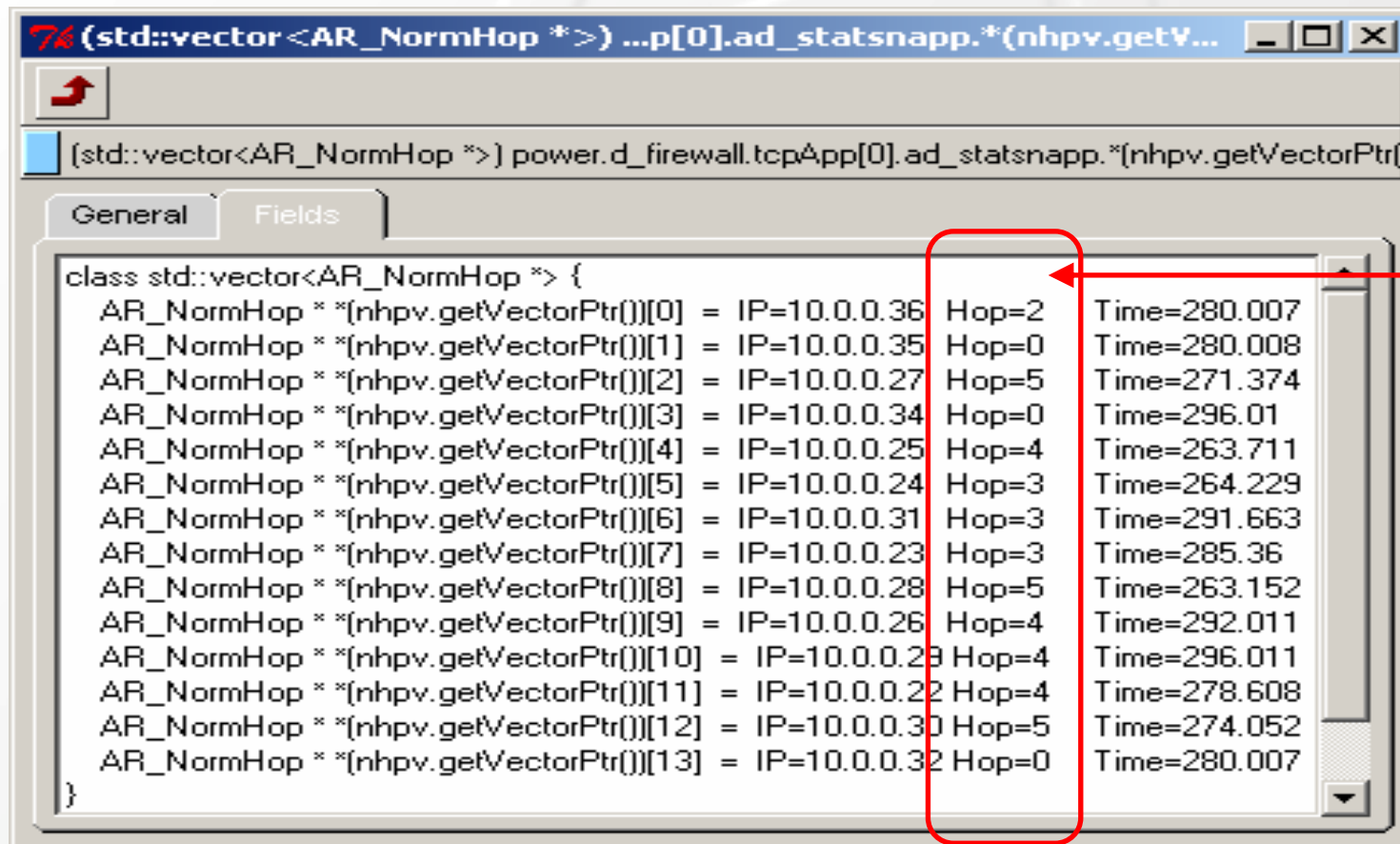


- Созданная среда моделирования позволяет проводить различные эксперименты с целью исследования стратегий реализации атак и перспективных методов защиты.
- В процессе экспериментов можно варьировать
 - топологию и конфигурацию сети;
 - структуру и конфигурацию команд атаки и защиты;
 - механизмы реализации атак и защиты;
 - параметры кооперации команд и др.



Режим обучения (1)

для Hop counts Filtering (HCF)



The screenshot shows a window titled "(std::vector<AR_NormHop *>) ...p[0].ad_statsnapp.*(nhpv.getV...". The window has two tabs: "General" and "Fields". The "General" tab is selected, displaying a list of nodes and their hop counts. A red box highlights the "Hop" column, and a red arrow points to it from the text "Количество скачков".

Node	Hop	Time
AR_NormHop * [nhpv.getVectorPtr()][0] = IP=10.0.0.36	Hop=2	Time=280.007
AR_NormHop * [nhpv.getVectorPtr()][1] = IP=10.0.0.35	Hop=0	Time=280.008
AR_NormHop * [nhpv.getVectorPtr()][2] = IP=10.0.0.27	Hop=5	Time=271.374
AR_NormHop * [nhpv.getVectorPtr()][3] = IP=10.0.0.34	Hop=0	Time=296.01
AR_NormHop * [nhpv.getVectorPtr()][4] = IP=10.0.0.25	Hop=4	Time=263.711
AR_NormHop * [nhpv.getVectorPtr()][5] = IP=10.0.0.24	Hop=3	Time=264.229
AR_NormHop * [nhpv.getVectorPtr()][6] = IP=10.0.0.31	Hop=3	Time=291.663
AR_NormHop * [nhpv.getVectorPtr()][7] = IP=10.0.0.23	Hop=3	Time=285.36
AR_NormHop * [nhpv.getVectorPtr()][8] = IP=10.0.0.28	Hop=5	Time=263.152
AR_NormHop * [nhpv.getVectorPtr()][9] = IP=10.0.0.26	Hop=4	Time=292.011
AR_NormHop * [nhpv.getVectorPtr()][10] = IP=10.0.0.29	Hop=4	Time=296.011
AR_NormHop * [nhpv.getVectorPtr()][11] = IP=10.0.0.22	Hop=4	Time=278.608
AR_NormHop * [nhpv.getVectorPtr()][12] = IP=10.0.0.30	Hop=5	Time=274.052
AR_NormHop * [nhpv.getVectorPtr()][13] = IP=10.0.0.32	Hop=0	Time=280.007

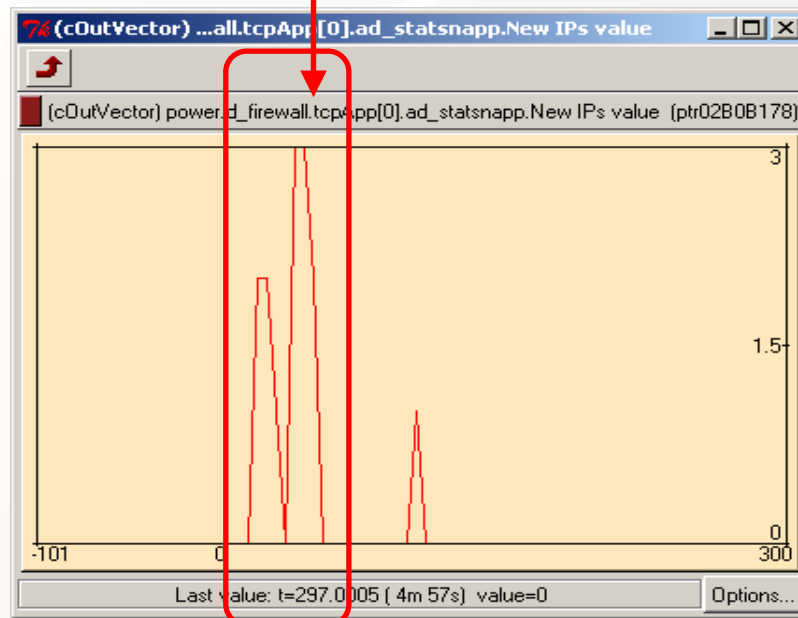
Коли-
чество
скачков

Список узлов, обращающихся к серверу, и скачков до них
после 300 секунд обучения

Режим обучения (2)

для Source IP address monitoring (SIPM)

Много новых адресов в начале



Изменение количества новых IP-адресов

Много новых адресов в интервале между 0 и 50 сек

The screenshot shows a list of new IP addresses and their associated times. A red rectangular box is drawn around the list, indicating a high number of new addresses in the 0-50 second interval. The list is as follows:

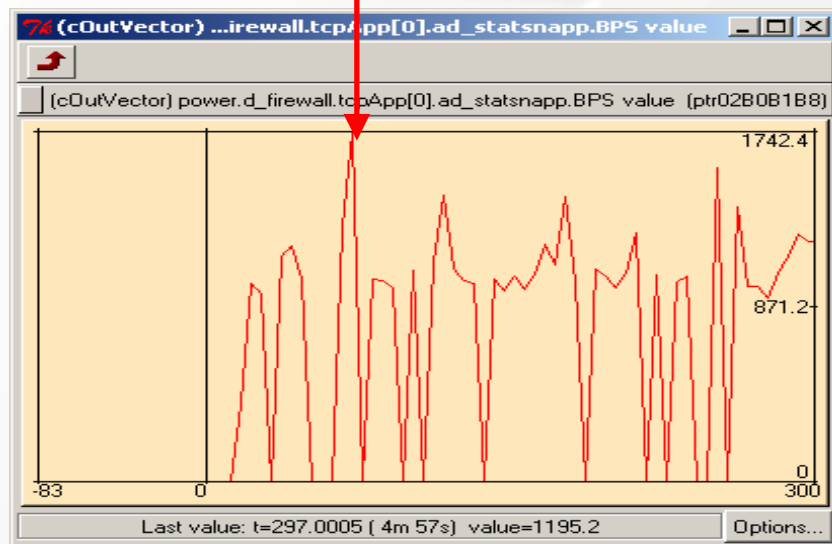
IP Address	Time
IP=10.0.0.33	Time=12.0008
IP=10.0.0.36	Time=14.0003
IP=10.0.0.37	Time=14.0001
IP=10.0.0.25	Time=23.1377
IP=10.0.0.27	Time=21.9945
IP=10.0.0.35	Time=21.9947
IP=10.0.0.24	Time=32.034
IP=10.0.0.23	Time=35.1563
IP=10.0.0.31	Time=33.8222
IP=10.0.0.28	Time=37.3439
IP=10.0.0.26	Time=39.8259
IP=10.0.0.29	Time=39.925
IP=10.0.0.22	Time=42.0896
IP=10.0.0.30	Time=45.5916
IP=10.0.0.32	Time=100.002

Список узлов, обращавшихся к серверу и признанных легитимными клиентами после 300 секунд обучения

Режим обучения (3)

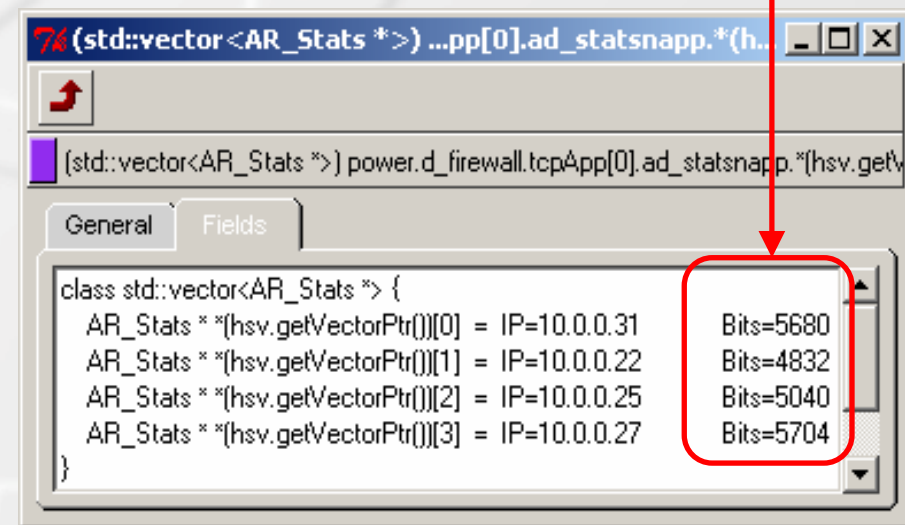
для Bit Per Second (BPS)

Максимальное значение -
1742.4 бит/сек



Изменение параметра BPS

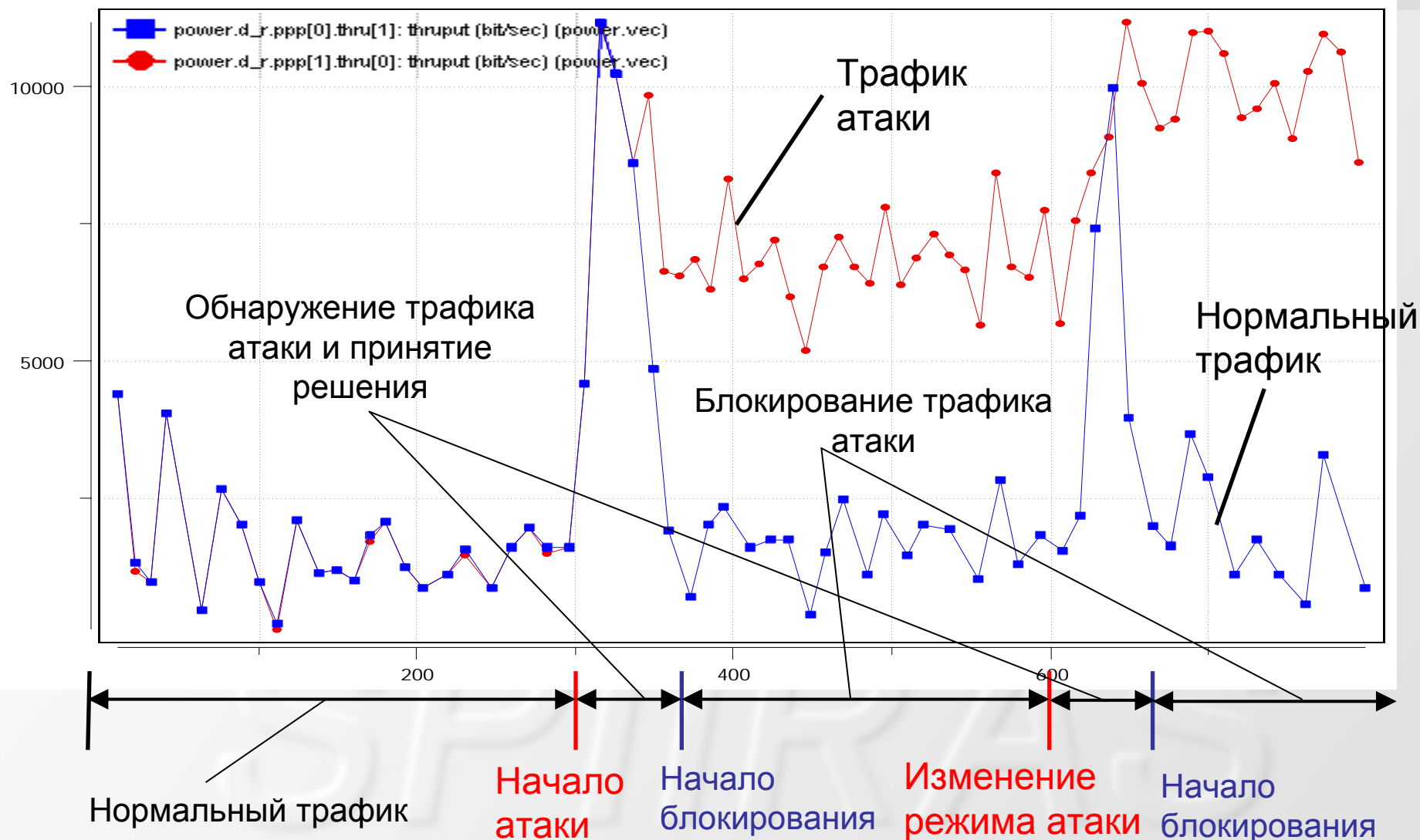
Количество бит в
интервале 10 сек



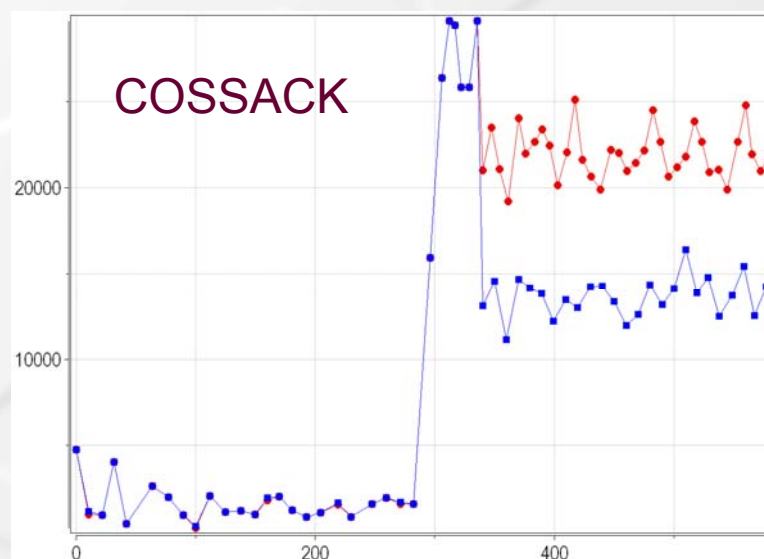
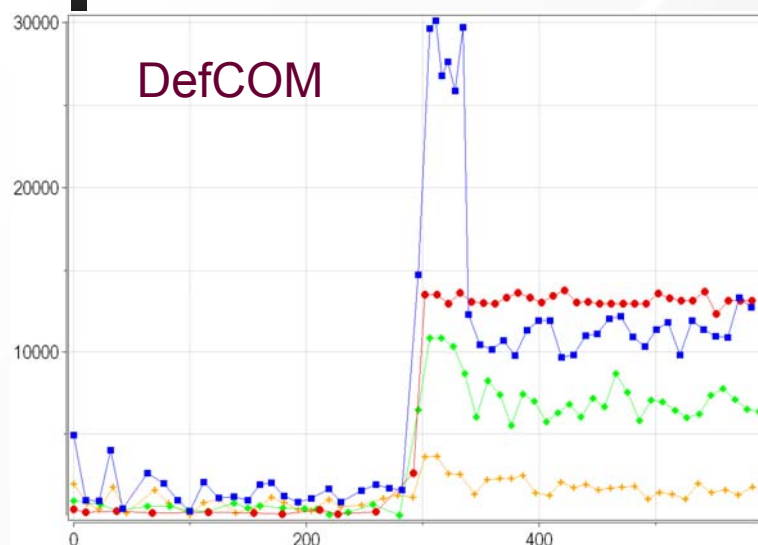
Статистика количества переданных
бит по одному из интервалов
времени

Принятие решения и функционирование

Графики изменения пропускной способности канала на входе в защищаемую сеть (зависимость бит/с от времени) до (красный) и после фильтра (синий)



Графики трафика для механизмов кооперации



- **Параметры:** топология опорной сети: $k=2$, количество узлов – 10, параметр $z = 2.25$, 10 клиентов подключены случайным образом к маршрутизаторам опорной сети, задан защищаемый сервер и параметры осуществления запросов к нему клиентов. В команду атаки входят 10 демонов, реализующих атаку UDP flood на сервер.





Основные результаты работы

- Разработана **программная среда** моделирования механизмов защиты от DDoS атак
- С использованием разработанной среды проведено множество **экспериментов**
- **Исследовались** различные типы атак, оптимальные параметров механизмов защиты
- Проведено **сравнение** различных механизмов защиты, режимов кооперации
- В дальнейшем **планируется** совершенствование среды моделирования, расширение библиотеки предметной области и системы многоагентного моделирования.



Контактная информация

Котенко Игорь Витальевич (СПИИРАН)

ivkote@iias.spb.su

<http://comsec.spb.ru/kotenko/>

Благодарности

Работа выполнена при финансовой поддержке РФФИ (проект №07-01-00547), программы фундаментальных исследований ОИТВС РАН (контракт №3.2/03), Фонда содействия отечественной науке и при частичной финансовой поддержке, осуществляемой в рамках проектов Евросоюза POSITIF (контракт IST-2002-002314) и RE-TRUST (контракт № 021186-2).



РОССИЙСКАЯ АКАДЕМИЯ НАУК

