New botnets trends and threats

André Fucs – afucs@uol.com.br
Augusto Paes de Barros – augusto@paesdebarros.com.br
Victor Pereira – vp@sekure.org

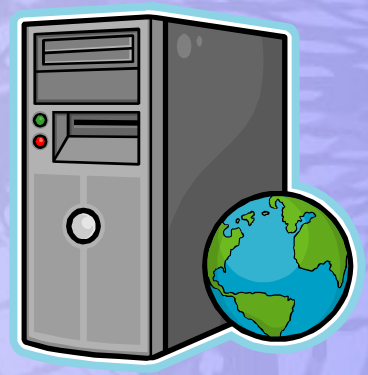
Agenda

- Botnet 101
- Botnet challenges
- A Layered approach
- Control layer
- Communication Layer
- Infection Layer
- Features Layer
- Back to the right side
- Conclusion

Botnet 101

"a collection of compromised machines running programs, usually referred to as worms, Trojan horses, or backdoors, under a common command and control infrastructure."

!!!!!!!!



!!!!!!!!

!!!!!!!!

!!!!!!!!

Botnet 101

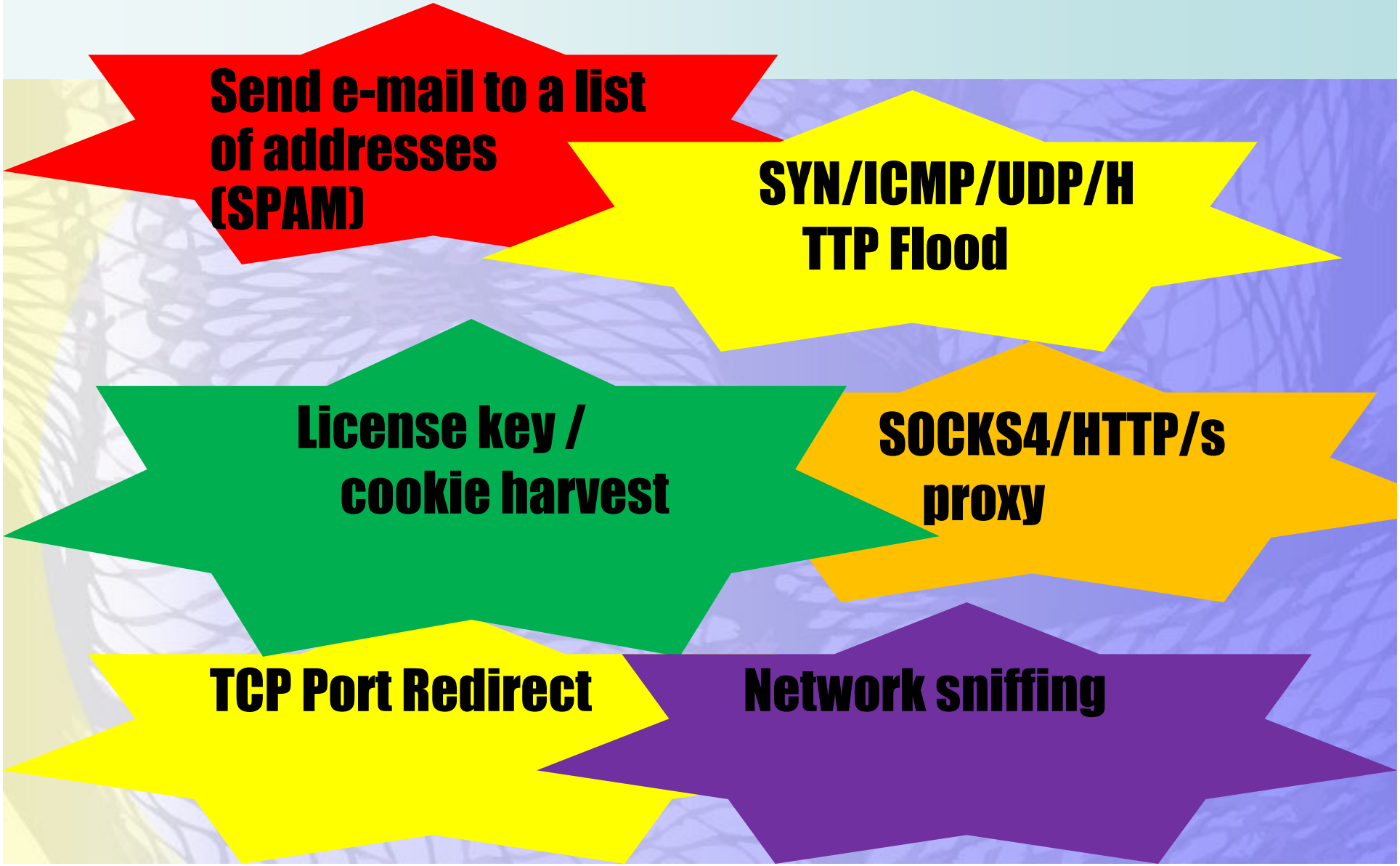
Botnets are an increasing threat:

The Dutch police found a 1.5 million node botnet

Telenor – Norwegian ISP – disbanded a 10,000 node botnet.

Botnet 101

Bots usually have limited feature set:



Botnet 101

- Botnets can deploy several control channels, still IRC is currently the most commonly used

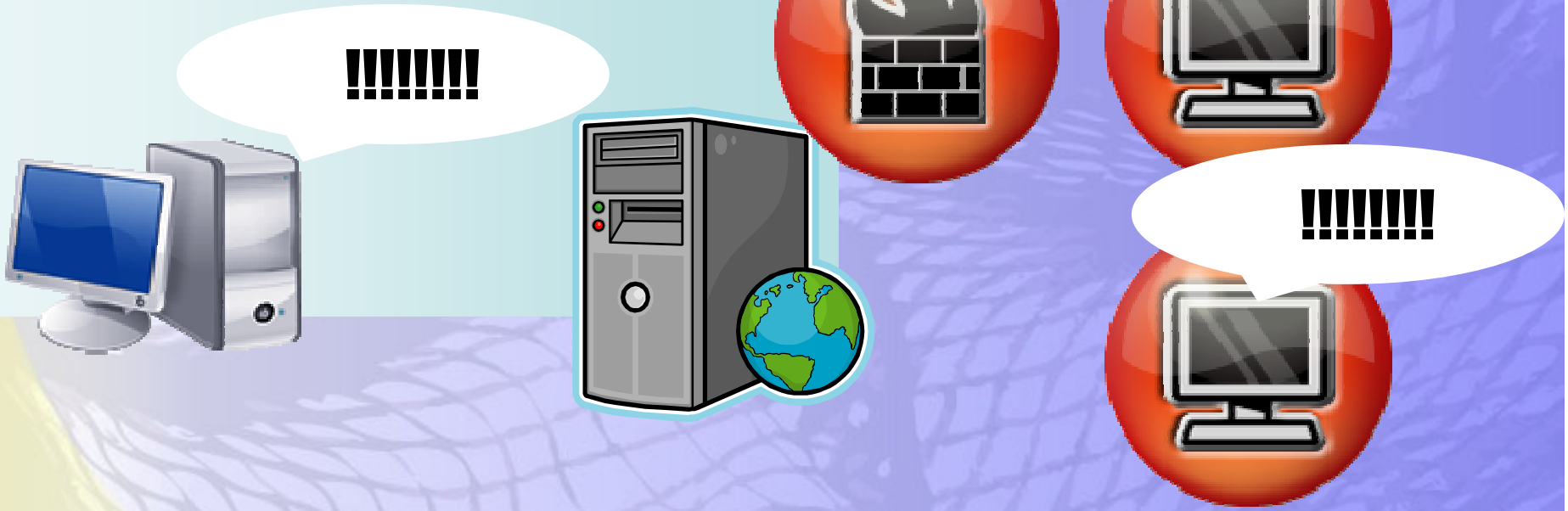
But IRC is not such a common protocol anymore…

At least not within corporate networks

Botnet challenges

- IRC issues:
  - Easy to block
  - Easy to be monitored

Web would be an easy choice, however…



Botnet challenges

- Increasing number of organizations are deploying content based screening…

  and it's easy to block….

- Second stage payload web sites are easy to track

  and easy to shutdown!

- But what about the home users?

Botnet challenges

The botnet MUST be able to infect new machines

Botnet MUST
download
stage

Botnet
download
second stage

Botnets must be smart P2P applications!

–The botnet MUST communicate and relay control messages to peer machines and be NAT trasversal capable

Botnet challenges

"In the traditional botnet, if you cut off the head, you kill the beast.

We speculate that, as more command-and control servers get identified by ISPs, you will see more and more of these botnets go to peer-to-peer."

Dean Turner, senior manager of development for Symantec.

A Layered approach

Or…

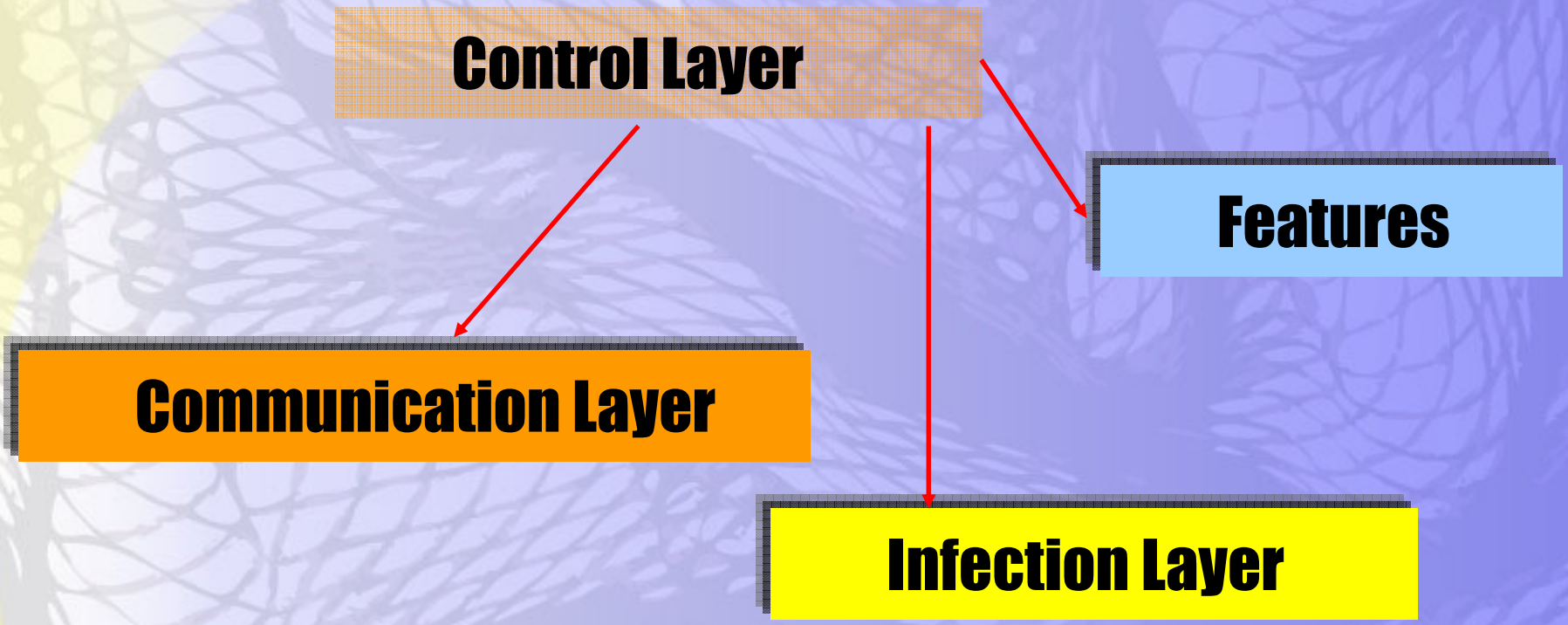
"Hello, may I speak with the product manager?"

A Layered approach

Why not build the bot in a way that:

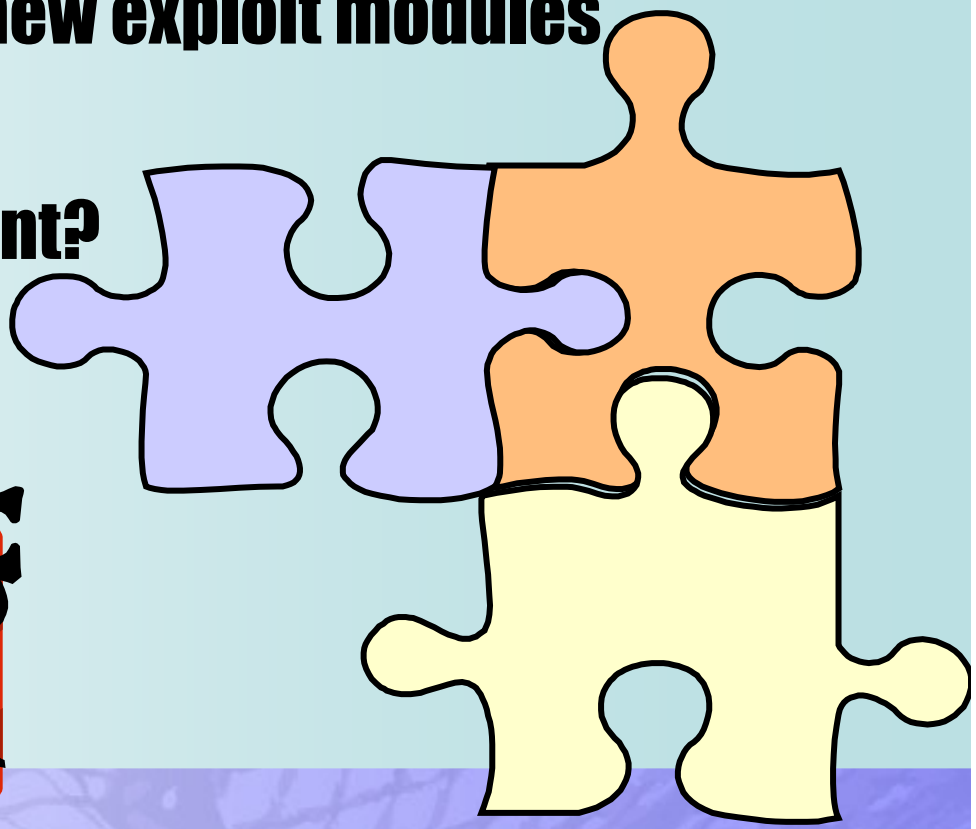
- You don't need to change the control logic when changing the communication protocol
- You can work with new features as plugins
- You can use different communication methods with the same basic code
- You don't need to release a new version when adding a new exploit
- You don't even need to code a new exploit!

A Layered approach



A Layered approach

- Why make it modular?
  - Possibility of infecting new machines without having to replace the whole bot – new exploit modules
  - Code re-use?
  - Lower cost of development?



A Layered approach

What about a botnet that has the following features:

- XML based communication;
- Secure control using digital signatures;
- Channel independent;
- Plug-in capable;

And even…

- .NET ready!

Control Layer

- Why an XML based control channel?
  - More or less easy to extend
  - Standard based
  - Amazing text based
  - Internet ready
  - Extremely pervasive
  - Easy to copy and paste on websites...

Control Layer

A bot should be small and deploy a minimum features as more advanced features should be either download or uploaded.

New features could be easily added to the bot

```
<command>
    <jobid>123</jobid>
    <feature id="module X">
    module parameters here
    </feature>
</command>
```

Control Layer

- Payload can be even more flexible
- Bot can simply receive VBS or IronPython code on a signed XML message and run it.

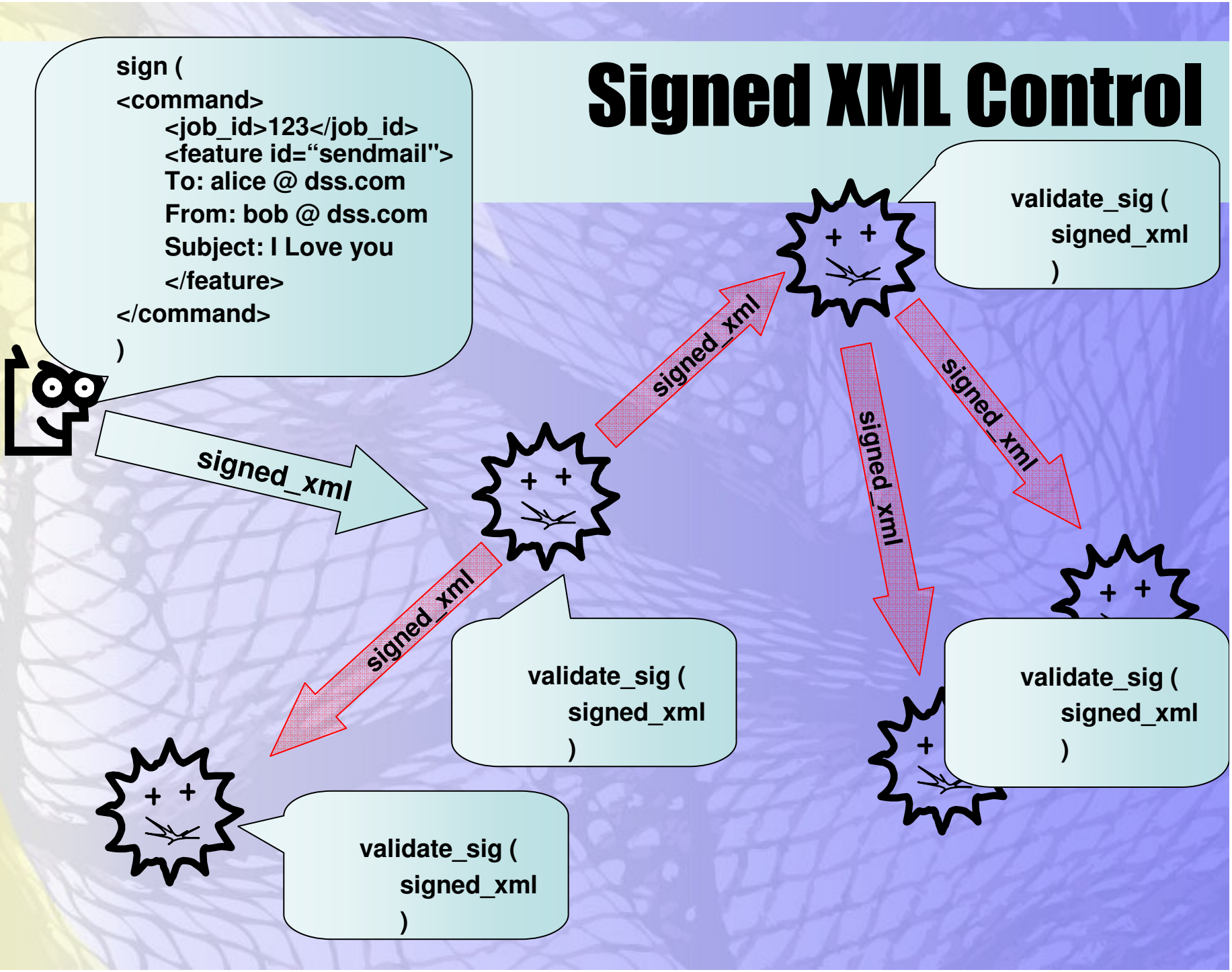
Both languages offer easy access to the .NET Framework

Scriptable bots!

Control Layer

- Why to use Digital signatures?
  - If we trust digital signatures to sign a dollar swap contracts, why shouldn't we digitally sign commands for a botnet?
  - Easy to implement, just think about XMLSIG...
  - May prevent botnet takeovers.

Signed XML Control



Communication and Control

- **Why seek Channel independence?**

  Imagine a world were:
  - **A botnet can download a payload from a web site;**
  - **Replicate the payload to another bot using different transports like:**
    - **Skype**
    - **SMB**
    - **SMS**
    - **SIP**
    - **RFC1149**
    - **…**

Communication and Control

Control:

- OTP based herder search
  - Helps to re-establish contact between bot herder and unpaired bots.
- Digital Signatures
  - Allows bot to replicate botnet commands to peer bots securely

Communication:

- Basic protocols covert channels
  - DNS, HTTP, 802.11
- P2P mechanism
  - Allows bot to communicate without herder intervention

OTP based herder search

- The bot always need to know how to reach its master
  - Really?
  - Reverse Engineering vulnerable
  - Found the herder location, game over
- What if the bot doesn't know where the herder is, but knows how to search for it?
- They need a shared secret
- The shared secret can't be static
- Isn't it just like the password dilemma?

OTP based herder search

Solution: One Time Passwords

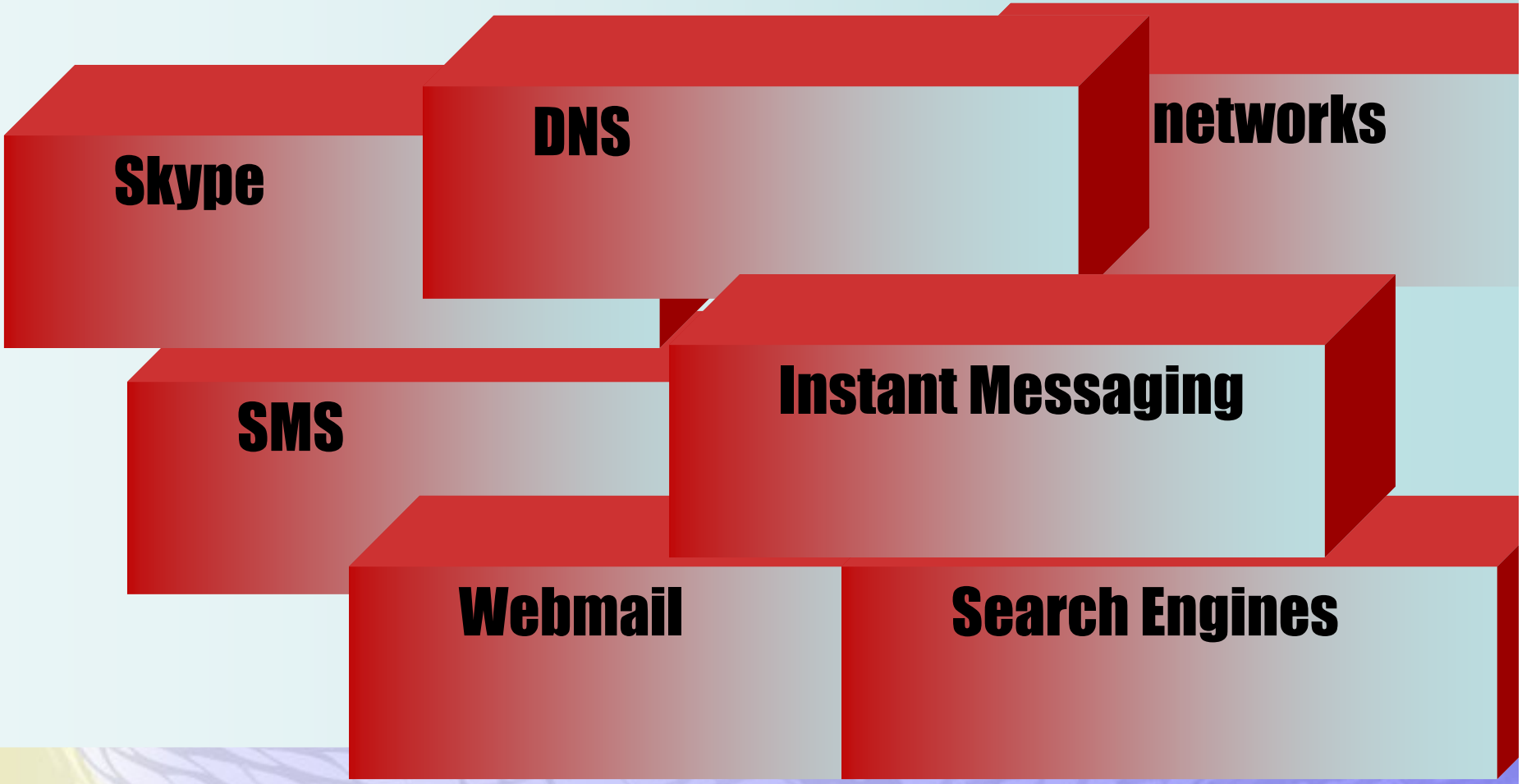
- Bot and herder have the same seed
- Both calculate a new OTP periodically
- Herder publishes information for the bot together with the OTP string
- Bot searches for the OTP string
  - On Google
  - On P2P networks
  - On Social Network Websites
  - Can search for a string posted by others? You can use it.

OTP based herder search

Demo: Using Skype Profiles

Communication Layer

A brief list of possible channels



Communication layer - Skype

Skype...

Pros

- Popular client
- P2P encrypted communication facilities
- NAT Friendly
- Firewall circumvention capabilities
- Easy to use API
- Profile Search capabilities

Cons

- Has Security Mechanisms to prevent unauthorized access to Skype client

Infection Layer

Why not embed something like MetaExploit to a bot?

- Exploits being published by others, ready for plug into the bot
- The framework as part of the bot
  - Just one payload – The bot
  - N exploits – How many available in Metasploit today?

Features Layer

DDoS, Spam,...

What else can a bot do?

- Criminals are making money by stealing users credentials for:
  - Auction sites
  - Online Banking

Source: Win32/Bancos – Malicious Software Encyclopedia
http://www.microsoft.com/security/encyclopedia/details.aspx?name=Win32%2fBancos

Features Layer

Those guys are improving their defenses:

- **Two-factor authentication**
  - **Tokens**
  - **OTP Cards / ‘ Bingo Cards’ – Very popular among Brazilian Banks:**

Features Layer

What a bot can do when two-factor authentication is being used?

- Transaction tampering is easy and hasn't been done until now…

Features Layer

Demo: Transaction Tampering on IE

Infection and feature nightmare

Let's go again on a "what if" scenario...

- One of the downloadable features is the packer/crypter used to build the bot
- A new bot can:
  - Rebuild itself with a new packer/crypter
  - Start spreading itself with new exploits
- AV nightmare!

Is it real? Is it possible?

Dr. Jose Nazario, from Arbor Networks, on Black Hat DC (3wks ago):

- ***Growing numbers of HTTP, IM and other bots***
- Ability of botnet herders is increasing
  - They will write their own communication protocols

- Last botnets studied show these trends are real
  - P2P is used by Storm Worm (01-2007)
  - HTTP is used by Korgo, Padobot, Bzub, Nuclear Grabber
  - Encryption – Nugache
  - Bots (Rbot, Sdbot, and Gaobot) compose three of the top five slots in terms of total number of removals (MSRT)

Back to the right side

“If a bad guy can persuade you to run his program on your computer, it's not your computer anymore”

Social Engineering is a key factor and a trend in terms of malicious software

Back to the right side

Now, more than ever, users should be prevented from running with administrative privileges – User training and awareness is key

Outbound traffic monitoring is still one of the few ways to detect bots in your network

Network Behavior Analysis may indicate the use of Covert Channels

Conclusion

- Botnets are growing and evolving fast but the are some things we can expect
  - They will be easily extended and upgraded
  - They wil traverse multiple types of network and protocols
  - Their master will not be easily found since not even the bot knows where to find him
  - They won't be easily hijacked as they only accept digitally signed commands
  - They will be able to directly change transactions made by users on websites and on-line banks, without needing to steal credentials
  - They will use as communication vectors protocols that can't be easily blocked without causing harm, like DNS and HTTP

Thanks

Aylton Souza, Emmanuel Gadaix, Gustavo Zeidan, Dr. Jose Nazario, Kenneth Chiedu Ogwu, Lincoln Moreira Junior

Cabral, for doing nothing

Ddos crew, for doing

And a special thanks to Paulo T.

References

- Paul Barford, Vinod Yegneswaran. An Inside Look at Botnets. Computer Sciences Department University of Wisconsin, Madison.
- Kelly Jackson Higgins, Senior Editor, *Dark Reading*. Black Hat: Botnets Go One-on-One. Feb, 2007.
- Evan Cooke, Farnam Jahanian, Danny McPherson. The Zombie Roundup: Understanding, Detecting, and Disrupting Botnets. USENIX SRUTI 05.
- Paul Bächer, Thorsten Holz, Markus Kötter, Georg Wicherski. Know your Enemy: Tracking Botnets. Using honeynets to learn more about Bots. Honeynet Project. March, 2005.
- David Dagon, Guofei Gu, Cliff Zou, Julian Grizzard, Sanjeev Dwivedi, Wenke Lee, Richard Lipton. A Taxonomy of Botnets. Georgia Institute of Technology, University of Central Florida.
- John Kristoff. Botnets. NorthWestern University. NANOG 32.
- Peter Judge. Computerworld Security, Cambridge professor warns of Skype botnet threat, January 25 2006.
- Dan Kamiksy. Black Ops of DNS. Black Hat USA, July 2004.
- Laurent Butti, Franck Veysett. Wi-Fi Advanced Stealh, Black Hat USA, August 2006.
- Dr. Jose Nazario. Botnet Tracking: Tools, Techniques, and Lessons Learned. Black Hat DC, March 2007.
- Microsoft. Malicious Software Removal Tool: Progress made, trends observed.
- Microsoft. Behavioral model of social engineering malware.