
Implementing the Virtual Area Network



John Newbigin

jnewbigin@swin.edu.au

7 April 2011



Implementing the VAN 1

Overview

Attempt to answer the following questions:

- Why do we need the VAN?
- How did it evolve?
- How does it work?
- What can you do with it?
- Where is it going in the future?



Implementing the VAN 2

In the beginning

- School of IT becomes Faculty of Information and Communication Technologies (circa 2005)
- Responsible for MCSE Labs
- Responsible for Cisco Labs
- Responsible for other labs where students require Administrator/root access



Implementing the VAN 3

The Problems

- Re-image labs several times per week/day
- Rouge DHCP servers
- Dual boot PCs
- No domain login
- NT4 (no USB)
- Unplugging "Uplink" cable



Implementing the VAN 4

The Idea

- VMware Player
- Linux
- Previous work at AIPC (VPN using ppp)
- Interest in Multicast

- No daily intervention
- Improve "Student Experience"

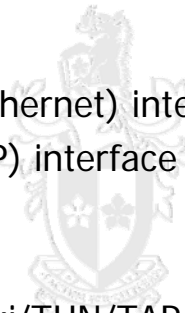


Implementing the VAN 5

Technology

TAP

- TAP is a virtual Layer 2 (ethernet) interface
- TUN is a virtual Layer 3 (IP) interface
- Native to Linux
- OpenVPN for Windows
- <http://en.wikipedia.org/wiki/TUN/TAP>
- Works well
- Comes with full source code



Implementing the VAN 6

Technology

Multicast

- Programmers don't understand multicast
- Network administrators don't understand multicast
- OS vendors don't understand multicast
- Unix Network Programming (Stevens) does understand multicast
- Properties of multicast are similar to ethernet



Implementing the VAN 7

Problems

- Obtaining IP Addresses/multicast groups
- Rate limits
 - Difficult to test with bulk data
- Firewalls
 - IGMP
- Split groups
 - Cisco problem?
- Poor performance with high utilisation



Implementing the VAN 8

Testing

The first use of the VAN for real work

- Build MCSE VPC images
- MCSE lab booked out
- SB202 available
- Random missing packets
- Wireshark
- Discovered Protocol Bindings
 - Able to configure full isolation

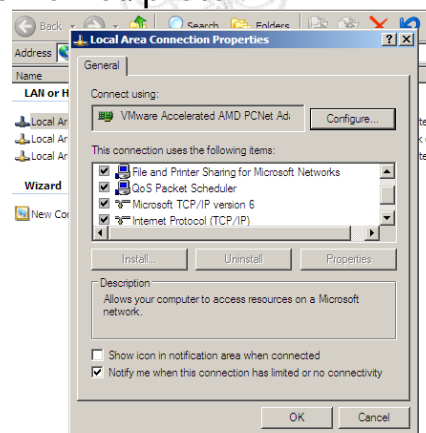


Implementing the VAN 9

Technology

Protocol Binder

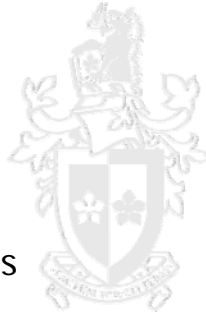
- <http://www.chrysocome.net/proto>
- MSYS
- COM Interface
- netsh



Implementing the VAN 10

Problem

- HET720 isolated network
- Can't re-image PCs
- Can't access Blackboard
- Can't save work
- Can't run any other classes
- Can't change room
- VMware images are large and hard to deploy



Implementing the VAN 11

Rejected Solution

LTPF Funding to:

- Put a 2nd PC on every desk
 - Use a KVM to save desk space
 - Install new network points & power
 - Install new switch blade
-
- Does not solve many of the problems



Implementing the VAN 12

Accepted solution

- Purchase more RAM
- Run VMware Player
- Purchase a server & storage
- Provide a method to download VMware images
- Connect VMware instances using the VAN
- Works across classrooms/buildings
- Solves all problems and more



Implementing the VAN 13

Technology

- wget / tar / bash script
 - Cross platform downloads
- cygwin for windows support
- Virtual Machine Launcher
 - GUI front end for windows
- High Speed Downloader
 - peer to peer download



Implementing the VAN 14

Problem

How can the virtual network get onto the physical network?

- 2nd network card
 - Not running TCP/IP
- Desk dongle
 - Make Console & Ethernet available on the desk
- Ready to change the Cisco world
- Wireless network cards
 - Eventually working



Implementing the VAN 15

Problem

More LTPF funding for PIX hardware

- Can we use the VAN instead of physical connections?
- Needs one VAN per ethernet interface
- Requires VNET identifier
- Port state information Up/Down
- Bridge VAN to VLAN for physical connections



Implementing the VAN 16

SwinGrid

- Need to run Globus in isolated environment
- 100 virtual machines in EN314
 - Hosted on 10 PCs
- 8 virtual machines on Green
 - Hosted on 2 nodes
- Communicate via VAN
- Route/NAT traffic to swinnet
 - Access to databases etc.



Implementing the VAN 17

AD209

- Allow access outside of scheduled classes
- Rack of 2811 routers
- Control access via web site
- Access ethernet via VAN & Virtual Machine Launcher
- Power on and off individual devices
- Solve usability problems
 - Mix routers & switches
 - Colour codes



Implementing the VAN 18

ATC Solution

- Less time setting up and packing up
- No moving heavy equipment
 - Safer
 - Less damage
- Less storage space required
- Able to use PoE, VoIP, Wireless
- Timetable flexibility
- Exam marking

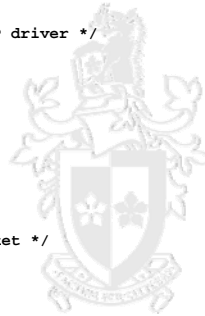


Implementing the VAN 19

How does it work?

```
/* This the packet data which comes from the TAP driver */
typedef struct
{
    unsigned char dest_mac[6];
    unsigned char src_mac[6];
    unsigned short ethertype;
    unsigned char payload[];
} tap_packet_t;

/* This is the payload we send over the UDP socket */
typedef struct
{
    unsigned char version; // Up to version 2
    unsigned char vnet; // like VLAN
    unsigned char fragment_id; // fragment counter
    unsigned char packet_id; // packet counter
    unsigned char host_id[2]; // lower 16 bits of IP address
    unsigned char payload[];
} virtual_ether_t;
```



Implementing the VAN 20

On the wire

Filter: `ip.dst_host == "239.200.201.100"` Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
2	0.498414	192.168.2.104	239.200.201.100	UDP	Source port: world-lm Destination port: cisco-sccp
10	5.949131	192.168.2.104	239.200.201.100	UDP	Source port: world-lm Destination port: cisco-sccp

Frame 2: 90 bytes on wire (720 bits), 90 bytes captured (720 bits)

Ethernet II, Src: VMware_Sf:c9:ba (00:0c:29:5f:c9:ba), Dst: IPv4mcast_48:c9:64 (01:00:5e:48:c9:64)

Internet Protocol, Src: 192.168.2.104 (192.168.2.104), Dst: 239.200.201.100 (239.200.201.100)

User Datagram Protocol, Src Port: world-lm (1462), Dst Port: cisco-sccp (2000)

Data (48 bytes)

Data: 020180d00268ffffffffffff00ff4900bbc7080600010800...

[Length: 48]

```

0000 01 00 5e 48 c9 64 00 0c 29 5f c9 ba 08 00 45 00  ..^H.d..)_....E.
0010 00 4c 95 4e 00 00 02 11 a7 15 c0 a8 02 68 ef c8  .L.N....h...
0020 c9 64 05 b6 07 d0 00 38 a8 ae 02 01 80 d0 02 68  .d.....8...
0030 ff ff ff ff ff ff 00 ff 49 00 b0 c7 08 05 00 01  .....I.....
0040 08 00 06 04 00 01 00 ff 49 00 b7 c7 a9 fe d1 39  .....I.....
0050 00 00 00 00 00 00 a9 fe 01 01
  
```

Version

VNET

Host ID

Packet ID

0x80 = Entire fragment

Payload is an APR packet

SWINBURNE SWINBURNE UNIVERSITY OF TECHNOLOGY

Implementing the VAN 21

How VNET works

- vnet works like the 802.1q VLAN Identifier
- No priority field
- Frames > 1500 octets don't fit inside 1500 octets
- UDP should not require fragmentation but...
- VAN will fragment to prevent UDP fragmentation

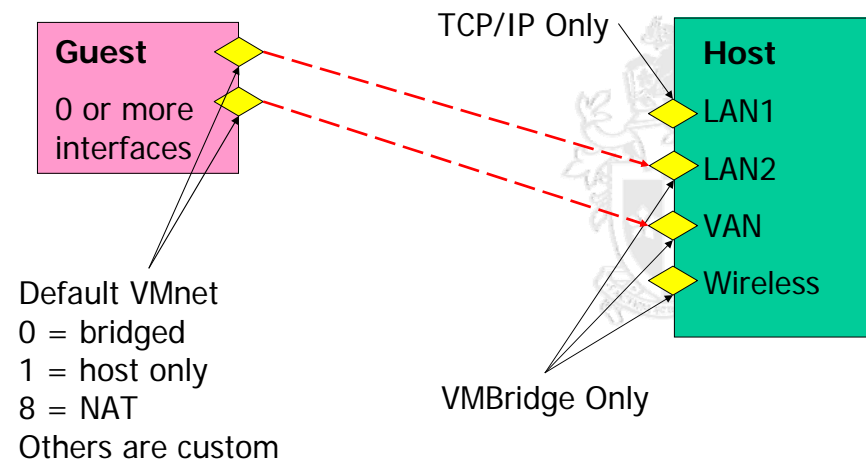
How VDP works

- Similar in concept to CDP
- Advertises existence and some state information
- `0x7f <hostname> | L=U`
 - Query switch via SNMP
- Virtual Machine Launcher shows a list of members
 - Hold time
- Ethernet state will reflect VDP link state



Implementing the VAN 23

VMware networking

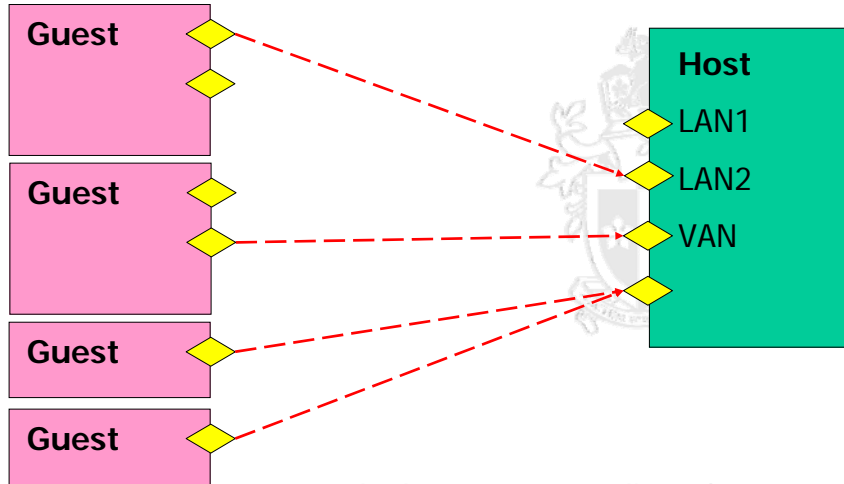


Up to 10 VMnets are available per Host



Implementing the VAN 24

VMware networking



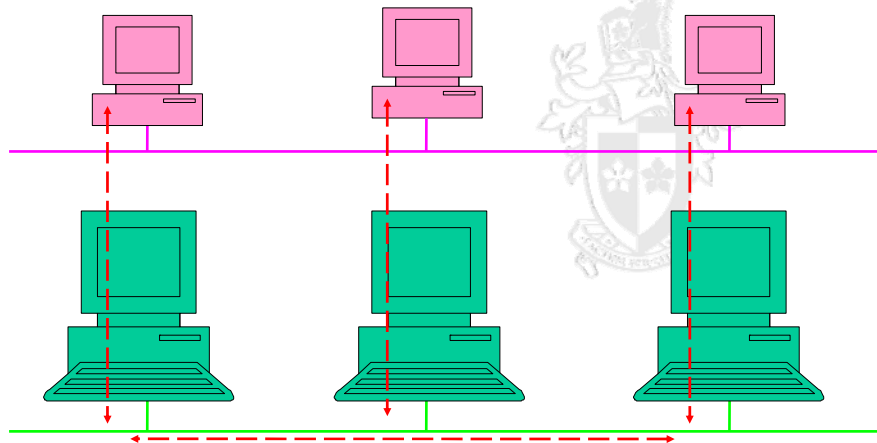
Multiple guests can talk to different interfaces or to each other



Implementing the VAN 25

Basic Topology

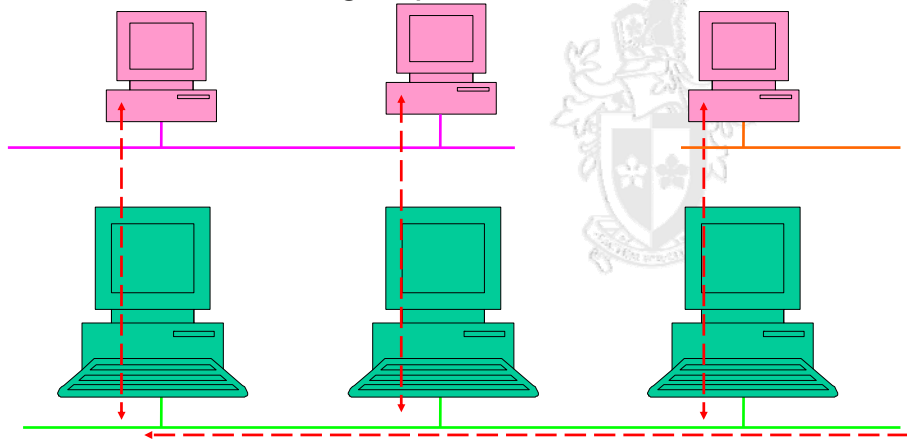
All VMs can talk to each other



Implementing the VAN 26

Multiple VNET Topology

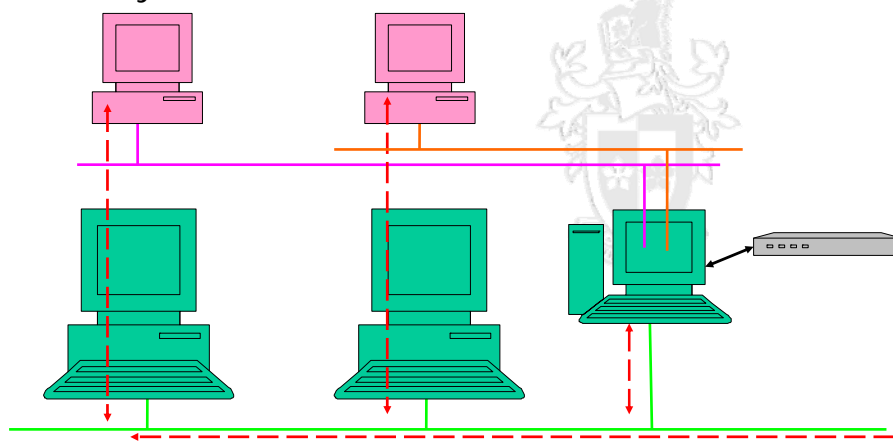
VMs can talk in groups



Implementing the VAN 27

VNET Server Topology

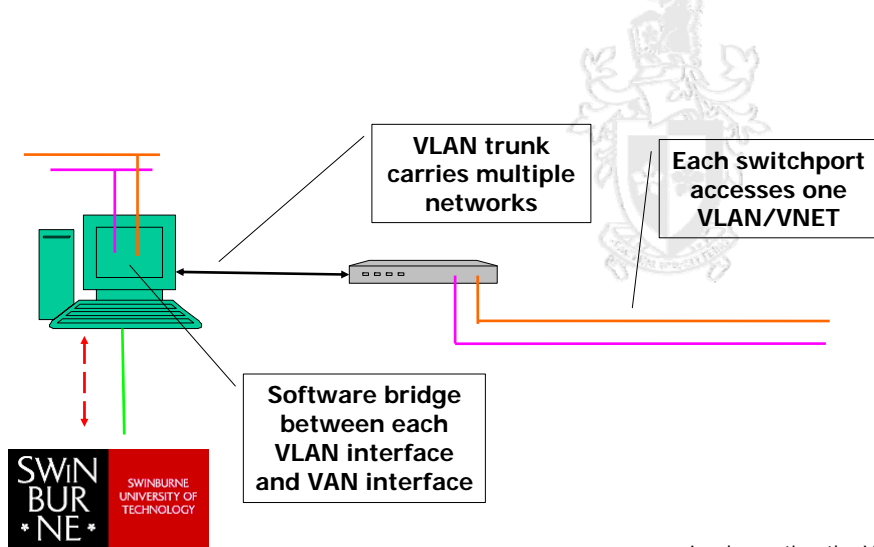
Every VM talks to the same server



Implementing the VAN 28

VNET Server Topology

The server talks to the Control Switch



Implementing the VAN 29

Future Possibilities

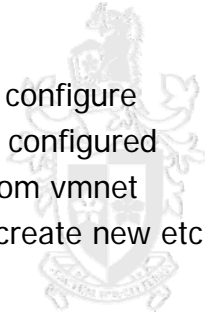
- Multiple VANs per Windows PC
 - Being a windows service there is no easy way to run multiple instances
- Change multicast IP address on the fly
- OS-X Support
- Access from home via VPN/unicast
- Virtual Switch
- VDP improvements
- Security



Implementing the VAN 30

Unsolved Problems

- VMware
 - vmnet bindings are hard to configure
 - Suspended images can't be configured
 - Limited UI controls for custom vmnet
 - UI has excessive features (create new etc.)
- Serial WIC cards

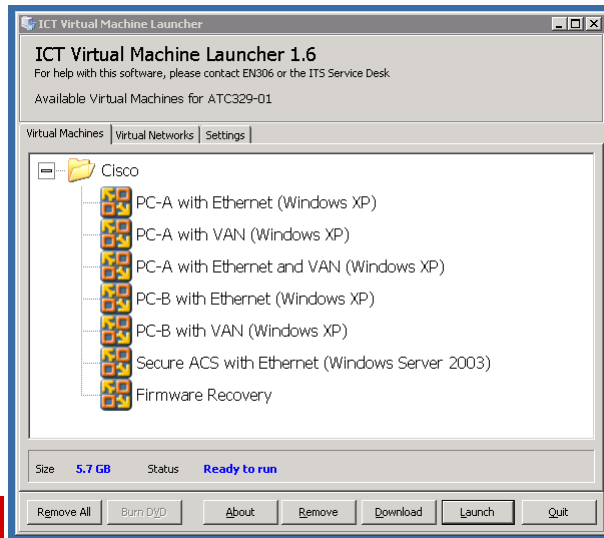


The End

- Questions?
- Hands on demonstration

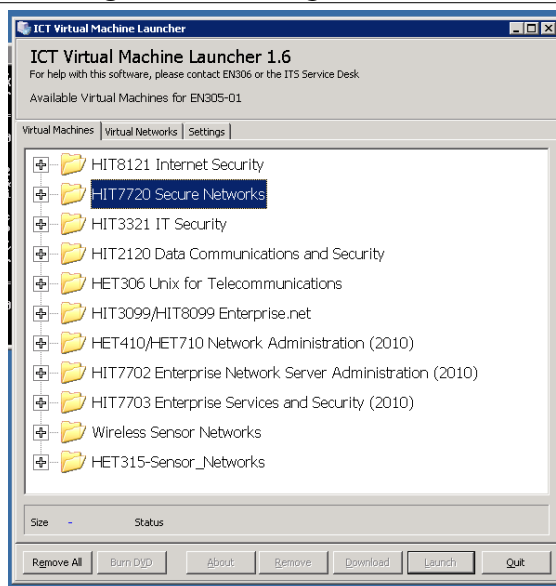


Cisco Virtual Machines



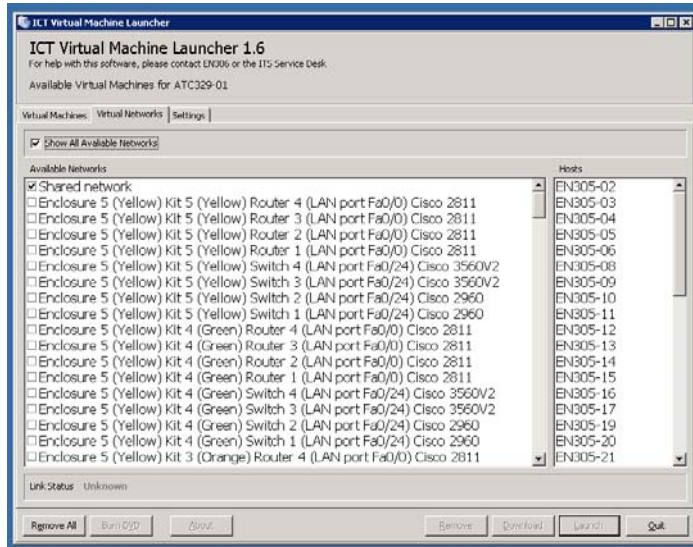
Implementing the VAN 33

Used by many ICT Subjects



Implementing the VAN 34

Virtual Network Selector



Implementing the VAN 35

High Speed Downloads

Image Name	Image Size	Number of files	Complete Copies	Incomplete Copies	Total Downloads
CentOS-5	3.9 Gb	9	3	0	3
Cisco	5.7 Gb	5	3	0	68
CiscoACS	2.4 Gb	6	3	0	30
CiscoB	3.6 Gb	6	3	0	21
Gordon	565.9 Mb	6	3	0	6
HET410-Windows7-InstallationLab	2.3 Gb	3	3	0	5
HET410-sWin08DC-2010	5.9 Gb	2	3	0	20
HET410-sWin08Svr-2010	5.6 Gb	4	3	0	44
HET410-sWin7PC1	5.5 Gb	2	3	0	20
HET410-sWin7PC2	5.5 Gb	2	3	0	20
HET710-2010_exam	12.8 Gb	9	3	0	3
HET720	4.3 Gb	5	3	0	27
HIT3321-Puppy-Linux	163.4 Mb	10	3	0	3
HIT3321-Win95	532.9 Mb	9	3	0	3
HIT3321-XPPPro-SP0_1_2	1.6 Gb	10	3	0	4
HIT3321-XPPPro-control_1_1	2.5 Gb	10	3	0	3
HIT3321-rh73_1_62	4.0 Gb	11	3	0	4
HIT7702-6421	22.3 Gb	1262	3	0	14
HIT7702-6430	24.8 Gb	964	3	0	5
HIT7703-6425	24.4 Gb	1606	3	0	3
HIT7703-6426	19.2 Gb	830	3	0	3
HIT7720-svr2k8	7.4 Gb	4	3	0	3
HIT8121-xp	4.6 Gb	5	3	0	4



Implementing the VAN 36