# Port set Type: Contiguous vs. Non-Contiguous

http://tools.ietf.org/html/draft-ietf-softwire-map-04

http://tools.ietf.org/html/draft-sun-dhc-port-set-option-00

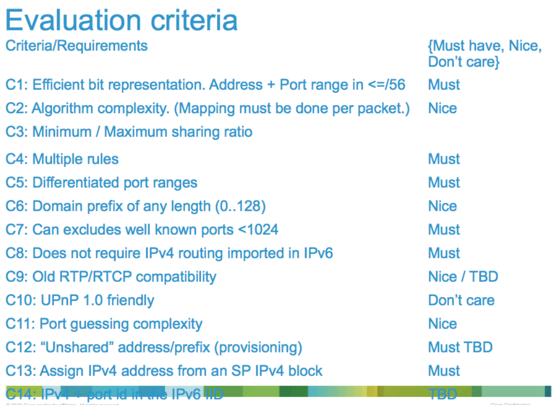
Qi Sun

2013.3 Orlando

# Motivation for port sharing

- IPv4 exhaustion
  - Several nodes share one IPv4 address by assigning non-overlapped port sets to each node
  - Providing IPv4 service without IPv4 routing on the provider IPv6 network
- Port set: Is contiguous port-set sufficient or do we need non-contiguous port-sets?

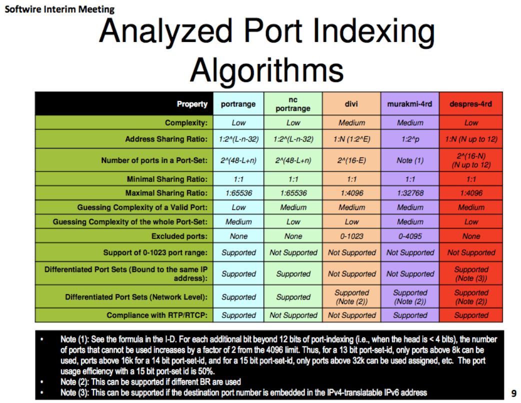
# Back in Beijing Interim Meeting



## Evaluation criteria

| Criteria/Requirements | {Must have, Nice, Don't care} |
|---|---|
| C1: Efficient bit representation. Address + Port range in <=/56 | Must |
| C2: Algorithm complexity. (Mapping must be done per packet.) | Nice |
| C3: Minimum / Maximum sharing ratio | |
| C4: Multiple rules | Must |
| C5: Differentiated port ranges | Must |
| C6: Domain prefix of any length (0..128) | Nice |
| C7: Can excludes well known ports <1024 | Must |
| C8: Does not require IPv4 routing imported in IPv6 | Must |
| C9: Old RTP/RTCP compatibility | Nice / TBD |
| C10: UPnP 1.0 friendly | Don't care |
| C11: Port guessing complexity | Nice |
| C12: "Unshared" address/prefix (provisioning) | Must TBD |
| C13: Assign IPv4 address from an SP IPv4 block | Must |
| C14: IPv4 + port id in the IPv6 IID | TBD |

From Ole's slides



Softwire Interim Meeting

## Analyzed Port Indexing Algorithms

| Property | portrange | nc portrange | divi | murakmi-4rd | despres-4rd |
|---|---|---|---|---|---|
| Complexity: | *Low* | *Low* | *Medium* | *Medium* | *Low* |
| Address Sharing Ratio: | *1:2^(L-n-32)* | *1:2^(L-n-32)* | *1:N (1:2^E)* | *1:2^p* | *1:N (N up to 12)* |
| Number of ports in a Port-Set: | *2^(48-L+n)* | *2^(48-L+n)* | *2^(16-E)* | *Note (1)* | *2^(16-N) (N up to 12)* |
| Minimal Sharing Ratio: | *1:1* | *1:1* | *1:1* | *1:1* | *1:1* |
| Maximal Sharing Ratio: | *1:65536* | *1:65536* | *1:4096* | *1:32768* | *1:4096* |
| Guessing Complexity of a Valid Port: | *Low* | *Medium* | *Medium* | *Medium* | *Medium* |
| Guessing Complexity of the whole Port-Set: | *Medium* | *Low* | *Low* | *Medium* | *Low* |
| Excluded ports: | *None* | *None* | *0-1023* | *0-4095* | *None* |
| Support of 0-1023 port range: | *Supported* | *Not Supported* | *Not Supported* | *Not Supported* | *Not Supported* |
| Differentiated Port Sets (Bound to the same IP address): | *Supported* | *Supported* | *Not Supported* | *Not Supported* | *Supported (Note (3))* |
| Differentiated Port Sets (Network Level): | *Supported* | *Supported* | *Supported (Note (2))* | *Supported (Note (2))* | *Supported (Note (2))* |
| Compliance with RTP/RTCP: | *Supported* | *Not Supported* | *Not Supported* | *Supported* | *Supported* |

- Note (1): See the formula in the I-D. For each additional bit beyond 12 bits of port-indexing (i.e., when the head is < 4 bits), the number of ports that cannot be used increases by a factor of 2 from the 4096 limit. Thus, for a 13 bit port-set-id, only ports above 8k can be used, ports above 16k for a 14 bit port-set-id, and for a 15 bit port-set-id, only ports above 32k can be used assigned, etc. The port usage efficiency with a 15 bit port-set id is 50%.
- Note (2): This can be supported if different BR are used
- Note (3): This can be supported if the destination port number is embedded in the IPv4-translatable IPv6 address

9

From Med's slides

Mainly focus on statelessly mapping IPv4 address and port into IPv6 prefix

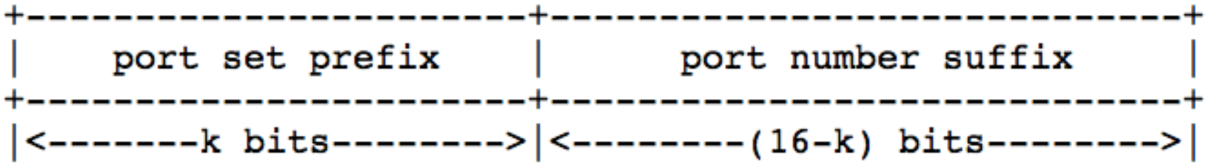
# Comparison Points

- Security
- Preservation of Well-Known Ports
- Complexity
- Backwards Compatibility with uPnP IGD:1

# Contiguous / Non-Contiguous Port Sets

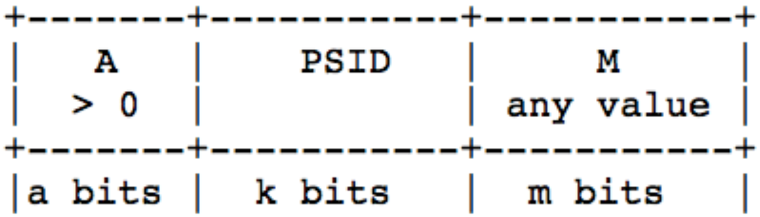
- Contiguous: A single port range per-client
- Non-Contiguous: Multiple port ranges distributed evenly across port space per-client

- Bit Presentation
  - Contiguous:



```
+-----------------------+------------------------------+
|    port set prefix    |      port number suffix      |
+-----------------------+------------------------------+
|<-------k bits-------->|<---------(16-k) bits-------->|
```

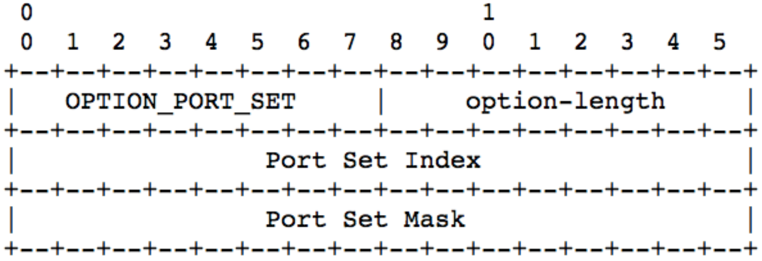
  - Non-Contiguous:



```
+-------+-----------+-----------+
|   A   |    PSID   |     M     |
|  > 0  |           | any value |
+-------+-----------+-----------+
|a bits |  k bits   |  m bits   |
```

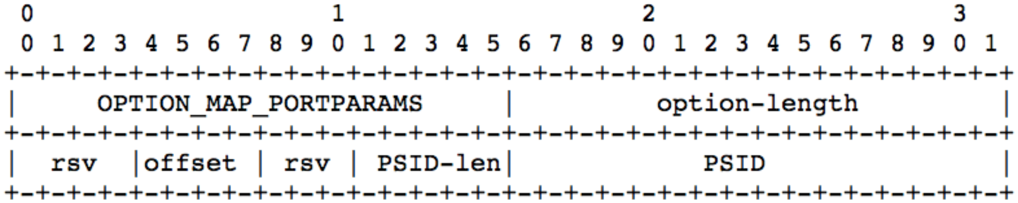
- Option format

Port Mask



```
 0                   1
 0  1  2  3  4  5  6  7  8  9  0  1  2  3  4  5
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|  OPTION_PORT_SET      |     option-length     |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                Port Set Index                 |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                Port Set Mask                  |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

GMA



```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     OPTION_MAP_PORTPARAMS     |         option-length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  rsv  |offset | rsv | PSID-len|             PSID              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

# Security

- Limited port range reduces port entropy -> it could be simpler for an attacker to guess ports
  - Source port randomization
- Ratio of address sharing increases -> the next port easier to predict
  - irrespective of whether it is contiguous or not
- Contiguous
  - Single port range: Predictable if allocation policy is known
- Non-Contiguous
  - Algorithmic port-set allocation: Predictable if allocation policy is known

# Preserving Well-Known Ports

- Contiguous
  - Don’t assign PSIDs falling within the WKP range
  - WKPs only available for the first few PSIDs
- Non-Contiguous
  - a-bits (A > 0)
  - PSID can be arbitrary, so that ISPs won’t be required to exclude some of prefixes (as PSID is part of MAP IPv6 prefix)
  - WKPs only available for the first few PSIDs

# Complexity

- Contiguous
  - Simple for CPE, Tunnel Concentrator, provisioning system, logging system, etc.
  - ‘Human readable’ format makes it easier to troubleshoot without tools
- Non-Contiguous
  - Brings complexity to all devices – CPE, server and clients (DHCP based)
  - Necessitates the use of tools to calculate allocated port ranges, complicating troubleshooting, logging, etc.
  - Could be hard to debug

# Backward Compatibility to uPnP

- Mainly about IGD:1
  - No external port negotiation
  - Fail if external port unavailable
- Testing shows neither have good compatibility
- Probability for IGD:1 to work normally is the same for both port-set algorithms

# Summary

|  | Contiguous Port-set | Non-Contiguous Port-set |
|---|---|---|
| Security | Predictable | Predictable |
|  | Sharing ratio increases -> Easier to predict (RFC[6056]) |  |
| Cost to Preserve WKP | Not allocate first few PSIDs | a-bit in port number (A > 0)<br>(PSID can be arbitrary) |
| Complexity | Low | High |
| Compatibility with IGD:1 | Not Good | Not Good |

- Non-contiguous port-sets offer little security with greater complexity.
- Conclusion: a simple contiguous port range, plus port randomization on the client side is preferable

# For the WG

- Is contiguous port-set enough?
- Conclusion?