NSCOM01

Protocol Standardization

2nd Term AY 2023-2024

Instructor: Dr. Marnel Peradilla

THE ROLE OF PROTOCOLS

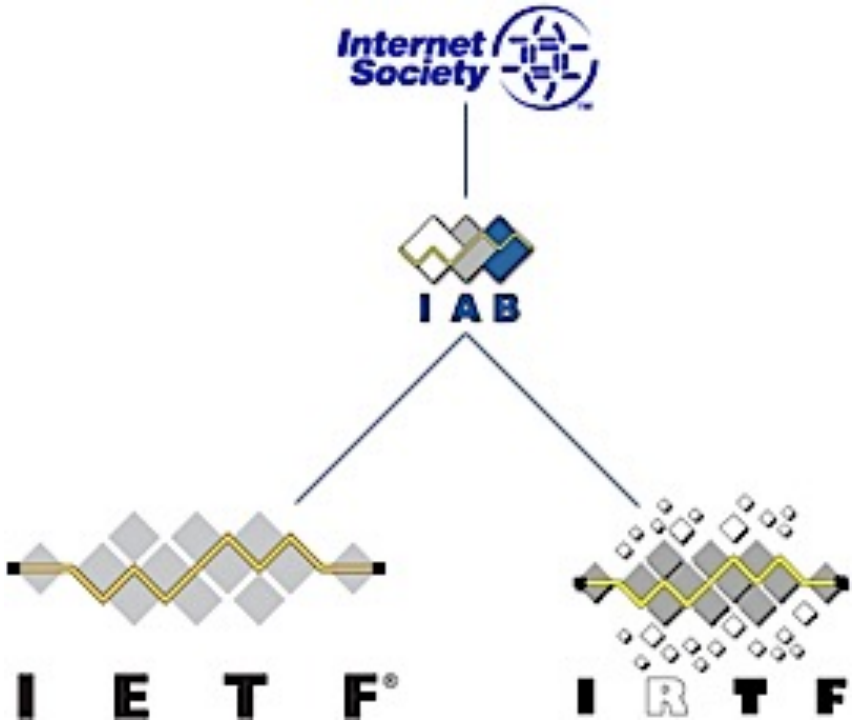
- **Networking protocols define a common format and set of rules for exchanging messages between devices.**
- **They govern different aspects of communication, such as**
  - Cable and connector designs
  - Message formatting
  - Message encoding
  - Transmission medium access and TX/RX timing
  - Available operations
  - Message exchange sequence
  - Error codes
  - etc

PROTOCOLS AND INDUSTRY STANDARDS

- **Well- known network protocols and communication devices conform to open standards in order to promote interoperability among different vendors / developers**
- **Standards organizations are usually vendor-neutral, non-profit organizations established to develop and promote the concept of open standards.**

INTERNET STANDARDS ORGANIZATIONS

- **Internet Society (ISOC)**
  - promotes open development and evolution of Internet use globally.
- **Internet Architecture Board (IAB)**
  - management and development of Internet standards.
- **Internet Engineering Task Force (IETF)**
  - develops, updates, and maintains Internet and TCP/IP technologies.
- **Internet Research Task Force (IRTF)**
  - focused on long-term research related to Internet and TCP/IP protocols.



ELECTRONICS AND COMMUNICATIONS STANDARD ORGANIZATIONS

- **Institute of Electrical and Electronics Engineers (IEEE)**
  - dedicated to advancing technological innovation and creating standards in a wide area of industries including networking.
- **Electronic Industries Alliance (EIA)**
  - standards related to electrical wiring, connectors, and network racks.
- **Telecommunications Industry Association (TIA)**
  - standards for radio equipment, cellular towers, Voice over IP (VoIP) devices, and satellite communications.
- **International Telecommunications Union-Telecommunication Standardization Sector (ITU-T)**
  - standards for video compression, Internet Protocol Television (IPTV), and broadband communications.

INTERNET STANDARDIZATION

❑ The RFC Process

- RFC stands for Request for Comments
- The RFC Series is a collection of documents containing informal memos, technical specs, etc on Internet technology
- Standardization on the Internet is usually achieved by building consensus through discussion of new technologies and protocols.
- An entity who would like to revise or propose a new protocol or technology publishes the proposal and requests others to comment on it
- RFCs are managed by the RFC Editor under the IETF

INTERNET STANDARDIZATION

❑ Types of RFCs

- Standards Track: RFC documents that describe technologies that are either already formally approved as standards, or they are likely to become standards in the future.
- Best Current Practice: A document providing guideline information or recommendations from the IETF that is not a formal standard.
- Informational: A document that provides general information or commentary.
- Experimental: A proposal for an experimental standard that is not considered to be on track to become an Internet standard- such as new protocols or proposed changes to existing protocols that were not accepted as formal standards
- Historic: Former standards that have been obsoleted.

INTERNET STANDARDIZATION

❑ Sources of RFCs

- IETF submissions
  - Sourced mostly from Working Groups or individual submissions via the IESG.
  - All are submitted to the RFC Editor by the IESG, after approval and with announcement to community.
- RFC Editor ("independent") submissions
  - Submitted directly to RFC Editor by outside entities (.e.g academic researchers, industry, etc).
  - IESG review for conflict with IETF activity, make publish/do-not-publish recommendation. RFC Editor has final decision, with advice from Editorial Board.
  - Classified only under Experimental or Informational categories

TOWARDS STANDARDIZATION

Internet Draft

- Initial submission of a document for review as a standard
- Usually revised many times based on feedback from IETF working groups

Proposed Standard

- An ID that is considered valuable, well-understood and stable
- Technology is mostly complete but may be revised based on further review, testing and experimentation

Draft Standard

- Technology was demonstrated to be functional on at least two independent and interoperable implementations.
- Specification is sufficiently mature, widely accepted and revised only if new issues arise

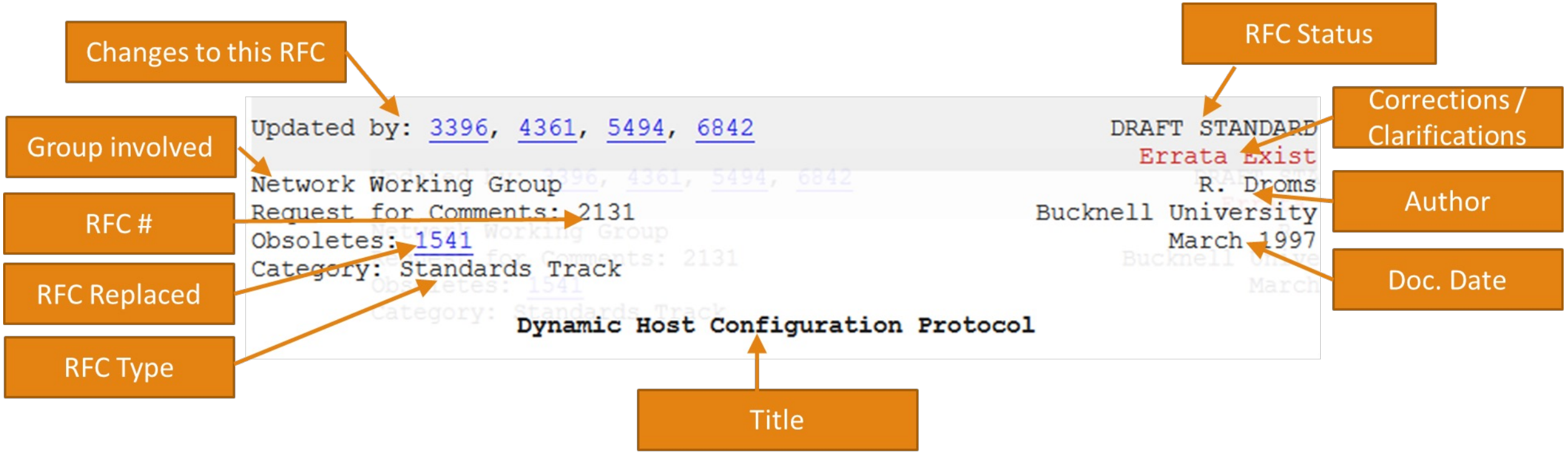
Internet Standard

- Very mature specifications that are popular and that have been widely implemented

RFC EDITOR INTERNET STANDARDS

https://www.rfc-editor.org/standards

READING AN RFC – THE RFC HEADER



READING AN RFC

Terminology	What it Means
MUST / REQUIRED / SHALL	Absolute requirement
MUST NOT / SHALL NOT	Absolute prohibition
SHOULD / RECOMMENDED	Item needs to be implemented, but can be ignored but only with valid reason and careful consideration of implications
SHOULD NOT / NOT RECOMMENDED	Item needs to be avoided, but can be acceptable with valid reason and careful consideration of implications
MAY	Optional, but must still interoperate with a system that implements the option

READING AN RFC

- **Abstract**
  - summary of the document
- **Document Body**
  - Introduction
  - Conventions - Use of terminology, abbreviations, etc
  - Main Text - describes algorithms used, message formats, error codes, etc
  - Security Considerations – discusses which attacks are considered by the protocol, which ones it is protected against, and which ones it is susceptible to including what can be done for defense
  - IANA Considerations – lists items for action by the Internet Assigned Numbers Authority when constants are used as part of protocol parameters

NOTATIONS ON STANDARDS DOCUMENTS

- **Protocol specifications commonly use a form of context-free grammar to specify syntax - usually for message or command formats**
- **A context-free grammar is a set of recursive rules used to generate string patterns**
- **On protocol specifications, the most popular context-free grammar notations used are:**
  - Backus-Naur Form (BNF)
  - Augmented Backus-Naur Form (ABNF)

BACKUS-NAUR FORM (BNF)

- **Every rule in Backus-Naur form has the structure: name ::= expansion**
  - The symbol `::=` means "may expand into" and "may be replaced with."
  - Every name is a non-terminal symbol and is surrounded by angle brackets, `< >`
    - `e.g. <animal>`
  - A terminal symbol is a literal like
    - `e.g. Shiba Inu`
  - A vertical bar `|` indicates choice
    - `e.g. <dog> ::= Shiba Inu | Labrador`
  - Juxtaposing expressions indicates sequencing.
    - `e.g. good <dog>`

BACKUS-NAUR FORM (BNF)

- **Every rule in Backus-Naur form has the structure: name ::= expansion**
  - Square brackets indicate an optional element
    - `e.g. hello [world]`
  - Commas indicate concatenation
    - `e.g. IP address := number, number, number, number`
  - An expansion is an expression containing terminal symbols and non-terminal symbols, joined together by sequencing and choice.
    - `e.g. good <dog> | bad <dog>`

BACKUS-NAUR FORM (BNF)

❑ Example

```
<animal>   ::= good <dog>   | bad <cat>
<dog>      ::= Shiba Inu    | Labrador
<cat>      ::= Persian      | Siamese
```

Based on the specified grammar:

`good Shiba Inu` ✓correct syntax

`bad Siamese` ✓ correct syntax

`good Persian` ✗ wrong syntax

BACKUS-NAUR FORM (BNF)

❑ Practice

- Formulate the BNF rules needed to define the syntax of a date using the notation Month Day Year (e.g. January 30 2020)

```
<date>   ::= <Month> <Day> <Year>
<Month>  ::= January|February|March|April|May
             |June|July|August|September|October
             |November|December
<Day>    ::= <digit>[<digit>]
<Year>   ::= <digit><digit><digit><digit>
<digit>  ::= 0 | 1 | 2 | 3 | 4 | 5 |6 |7 | 8 | 9 |
```

AUGMENTED BNF (ABNF)

- **Augmented Backus-Naur Form is a modified version of BNF that has changes in naming rules, repetition, alternatives, order-independence, and value ranges**
- **ABNF rules take on the form name = elements crlf**
  - `<name>` is the name of the rule,
  - `<elements>` is one or more rule names or terminal specifications
  - `<crlf>` is the end-of-line indicator (carriage return followed by line feed)
- **Terminal symbols can be specified directly as strings and are considered case insensitive**
  - e.g `Course = “NSCOM01”`
- **Comments are indicated using a semicolon (“;”) after the rule**
  - e.g. `Course = “NSCOM01” ; Network Applications`

AUGMENTED BNF (ABNF)

- **Terminal symbols can also be specified directly as strings or by one or more numeric characters, with the base interpretation of the characters indicated explicitly.**
  - b = binary
  - d = decimal e.g. ‘A’ = %d65
  - x = hexadecimal e.g. ‘A’ = %x41
- **A concatenated string of characters specified using numeric values uses the period (“.”) as separator**
  - e.g. ABC = %d65.66.67

ABNF CORE RULES – RFC 5234

```
ALPHA   =  %x41-5A / %x61-7A          ; A-Z / a-z
BIT     =  "0" / "1"
CHAR    =  %x01-7F                    ; any 7-bit US-ASCII character, excluding NUL
CR      =  %x0D                       ; carriage return
CRLF    =  CR LF                      ; Internet standard newline
CTL     =  %x00-1F / %x7F             ; controls
DIGIT   =  %x30-39                    ; 0-9
DQUOTE =  %x22                        ; " (Double Quote)
HEXDIG  =  DIGIT / "A" / "B" / "C" / "D" / "E" / "F"
HTAB    =  %x09                       ; horizontal tab
LF      =  %x0A                       ; linefeed
OCTET   =  %x00-FF                    ; 8 bits of data
SP      =  %x20                       ; space
VCHAR   =  %x21-7E                    ; visible (printing) characters
WSP     =  SP / HTAB                  ; white space
```

ABNF

❑ ABNF Operators:

1. Concatenation: Rule1 Rule2
   - e.g. `foo = “NSCOM”`
   - `bar = “01”`
   - `mumble = foobar ; NSCOM01`
2. Alternative: Rule1 / Rule2
   - e.g. `foo = “NSCOM0”`
   - `bar = “1” / “2” / “3”`
   - `mumble = foobar ; accepts NSCOM01, NSCOM02 and NSCOM03`
3. Value Range Alternatives: %c##-##
   - e.g. `foo = “NSCOM0”`
   - `bar = %x31-33`
   - `mumble = foobar ; accepts NSCOM01, NSCOM02 and NSCOM03`

ABNF

❑ ABNF Operators:

4. Sequence Group: (Rule1 Rule2)

Elements enclosed in parentheses are treated as a single element

```
e.g.  foo    =  "NSCOM01"
      bar    =  "NSSECUR"
      mumble =  "studying" (foo / bar) "now"   ; studying NSCOM01 now or studying NSSECUR now
```

5. Alternative: Rule1 / Rule2

```
e.g.  foo    =  "NSCOM0"
      bar    =  "1" / "2" / "3"
      mumble =  foobar              ; accepts NSCOM01, NSCOM02 and NSCOM03
```

6. Value Range Alternatives: %c##-##

```
e.g.  foo    =  "NSCOM0"
      bar    =  %x31-33
      mumble =  foobar              ; accepts NSCOM01, NSCOM02 and NSCOM03
```

BACKUS-NAUR FORM (BNF)

❑ Practice

- Given the command syntax of the File Transfer Protocol specified below using BNF,

```
<command> ::= <operation> <SP> <pathname> <CRLF>
<operation>:= STOR | RETR
<pathname> ::= <string>
<string> ::= <char> | <char><string>
<char> ::= any of the 128 ASCII characters except <CR> and  <LF>
```

1. *Is a single-character pathname allowed?*
2. *Is this pathname allowed? : "Hello/World/Hello World.txt"*
3. *Give the sequence of commands that stores a file named "practice.doc" and retrieves it immediately after*

ABNF

❑ Practice

- Formulate the ABNF rules needed to define the syntax of a date using the notation Month Day, Year (e.g. January 30, 2020)

```
date = <Month> SP <Day> “,” SP <Year>

Month = “January”/ “February”/”March”/”April”
        /”May”/”June”/”July”/“August”/”September”
        /”October”/“November”/“December”

Day = 1*2DIGIT

Year = 4DIGIT
```

MESSAGE FROM DPO

"The information and data contained in the online learning modules, such as the content, audio/visual materials or artwork are considered the intellectual property of the author and shall be treated in accordance with the IP Policies of DLSU. They are considered confidential information and intended only for the person/s or entities to which they are addressed. They are not allowed to be disclosed, distributed, lifted, or in any way reproduced without the written consent of the author/owner of the intellectual property."