Exchanging Microsoft: Implementing 27+ MS Exchange Protocols & APIs in OSS with grommunio

Jan Engelhardt, Lead Software Architect
MAPI
Messaging Application Programming Interface

...it's a slightly ambiguous term.
Used for:

• Concepts and the data model
• Programming interfaces
• Network protocols on the wire
Leading MAPI Products

Microsoft Exchange
Exchange Online
Microsoft Exchange Protocols & Specifications

The specifications are quite vast.


132+ documents, 8400+ pages.

And that is on top of the Internet Mail protocols (i.e. RFC 5322, 5545, etc.) that need to be supported anyway.
Leading Open Source Project

gromox by grommunio
### EX/MAPI related:
- OXCDATA
- OXDSCI
- OXCFOLD
- OXCXICS
- OXCICAL
- OXCMAIL
- OXCMAPIHTTP
- OXCMG
- OXCNOTIF
- OXNSPI
- OXPERM
- OXPRPT
- OXCROS
- OXCRPC
- OXCSTOR
- OXCTABL
- OXMSG
- OXOABK
- OXOABKT
- OXOCAL
- OXOCNCT

### Protocols and formats in gromox:
- OXODLGT
- OXOMSG
- OXORULE
- OXOSFLD
- OXOSMIME
- OXPROPS
- OXTNEF
- OXVCARD
- OXABREF
- OXOCFG
- OXRTPCP
- OXWAVLS
- OXWOOF
- OXWCDATA
- OXWSCONT
- OXWSFOLD
- OXWSMG
- OXODLGT
- OXWSMTGS
- OXWSPOST
- OXWSRSLNM
- OXWSJSON
- OXWSYNC
- OXSTASK
- PST
- CFB
- RCPE
- RPCH
- DCERP/C706
Tools for protocol analysis: MAPI/ROP

Decoding

• SSL intercept, decrypt, parse with Fiddler / Wireshark
• confer/compare with specs to validate
• write marshalling code

Clients

• MFCMAPI
• Outlook
Pinpointing issues

- User actions often entail multiple remote procedure calls.
- Request logging at e.g. the HTTP, DCERPC or ROP layers can help narrow down logic issues - (e.g. when packet framing is right but the content is not as expected)

```
[1/5] rop_dispatch(ropGetContentsTable) EC=ecSuccess RS=ecSuccess
[2/5] rop_dispatch(ropSetColumns) EC=ecSuccess RS=ecSuccess
[3/5] rop_dispatch(ropSortTable) EC=ecSuccess RS=ecSuccess
[4/5] rop_dispatch(ropSeekRow) EC=ecSuccess RS=ecSuccess
[5/5] rop_dispatch(ropFindRow) EC=ecSuccess RS=ecSuccess
```

Entering nsp_interface_get_specialtable: {container=0h record=0h delta=0 fpos=0/50}
Leaving nsp_interface_get_specialtable: {container=0h record=0h delta=0 fpos=0/50} {#outrows=2}
#0 (6 props)
  #0 ~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~
#1 (6 props)
#1 ~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~@~
6-e525584~@
```
[1/4] rop_dispatch(ropFindRow) EC=ecSuccess RS=ecNotFound
[1/5] rop_dispatch(ropFindRow) EC=ecSuccess RS=ecNotFound
C[1/5] rop_dispatch(ropFindRow) EC=ecSuccess RS=ecNotFound
```
Tools for protocol analysis: EWS

- observe Outlook etc. sending EWS requests (let gromox dump the XML)
- replay requests to an Exchange Server
- confer/compare with the specifications
- make Gromox respond to the EWS request

Clients

- Outlook for Mac
- MacMail
- Evolution (partially)
- Postman / curl
POST https://srv-exchange-0

POST https://srv-exchange-0

POST https://ews.url

Params  Authorization  Headers  (0)  Body  Pre-request Script  Tests  Settings

- none  form-data  x-www-form-urlencoded  raw  binary  XML

```
<?xml version="1.0" encoding="UTF-8"?>
  <soap:Header/>
  <msex:GetItem xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <itemShape>IdOnly</itemShape>
    <bodyType>HTML</bodyType>
    <additionalProperties />
    <fieldUri FieldUri="item:Sensitivity"/>
  </msex:GetItem>
</soap:Envelope>
```

Body  Cookies  Headers  Test Results

Pretty  Raw  Preview  Visualize  Text  View

Status: 200 OK  Time: 161 ms  Size: 4.56 KB  Save Response
Fixing specification errors


Contributions to specs
Jan Engelhardt edited this page on Dec 14, 2023 · 7 revisions

Instances where Microsoft documentation was updated.

- **PT_LONG is not for the unsigned**
- **Outlook invokes unspecified behavior with regard to PR_SENSITIVITY**
- **Fix size mentions for named properties**
- **Mention EX2019 behavior with regard to SPropertyRestriction**
- **MAPI_HARD_DELETE not fully documented**
- **relationship between DIR_ENTRYID and CONTAB_ENTRYID**
- **EWS: Month can have value 0 as well**
- **EWS: Default property table with bogus entry**
Future of gromox

• Better parallelization of accessing one and the same message store
• improved support for Internet Mail specs
• Graph API (See https://github.com/grommunio/grommunio-next)
• Continuing reporting spec errors/omissions (8400+ pages are not always right or complete)
Thanks!

Questions?