



17.5.2 Repositório Remoto de arquivos

O repositório Remoto é um sistema de arquivos gerenciado pelo IED. Portanto, visto que a norma não estabelece regras para nomenclatura dos arquivos e nem mesmo para sua a gerência por parte do IED, a lógica de funcionamento do ftp61850 pode ser afetada por tais questões, conforme esclarecido nos próximos parágrafos.

Para definir o diretório remoto onde o SAGE procura periodicamente por novos arquivos, deve-se usar o *token* FTD= da CNF. Se o diretório especificado utilizar a sintaxe "FTD= /", o ftp61850 não irá concatenar essa especificação de diretório aos nomes dos arquivos nos pedidos de FileOpen porque, nesse caso, o SAGE obtém, nas respostas para os pedidos de FileDirectory, o caminho completo dos arquivos de oscilografia a serem lidos.

17.5.3 Funcionamento do Processo

O processo ftp61850 depende apenas dos processos MCAST, GMCD, do conversor I61850 e do transportador de protocolo MMS. Além disso, para que ele seja iniciado com sucesso, é necessário que haja pelo menos uma CNF configurada para aquisição em protocolo IEC/61850 possuindo no seu final os *tokens* FTP, FTJ e FTD, conforme mostrado no exemplo a seguir:

CNF

```
CONFIG= ApTitle= ..... etc .... FTP= 120 FTJ= 3 FTD= dr_unextracted/
```

O *token* FTP define a periodicidade (segundos) de varredura do IED em busca de novos arquivos de oscilografia. No exemplo acima, o processo ftp61850 faria uma busca por novos arquivos a cada 120 segundos. No momento da busca, caso um ou mais arquivos sejam encontrados no diretório determinado pelo *token* FTD, eles são lidos para o **repositório local de oscilografias** do IED. Após o *download*, o arquivo não é apagado do repositório remoto, ou seja, assume-se que o IED se encarregará de apaga-lo após o término da transferência.

No ciclo seguinte de downloads (FTP segundos depois), caso o arquivo lido (mesmo nome) ainda esteja no repositório remoto, três situações podem ocorrer:

1) O arquivo é desprezado.

→ Este caso ocorre quando a data de modificação do arquivo remoto é igual a data de modificação do arquivo local. Assume-se que é o mesmo arquivo.

2) O arquivo é lido novamente sobreescrevendo o que já havia sido lido na varredura anterior.

→ Este caso ocorre quando as seguintes condições são satisfeitas simultaneamente (AND lógico)

- i. O time-tag mudou (mais novo)
- ii. O arquivo aumentou de tamanho
- iii. A diferença entre os time-tags é < FTPJ

Obs: este caso foi considerado para prevenir situações onde o arquivo que foi lido ainda não havia sido completamente gravado pelo IED, ou seja, ainda estava sendo incrementado com informações oscilográficas. Neste caso, deve ser lido novamente substituindo o arquivo incompleto no repositório local.

3) O arquivo é lido novamente como novo arquivo.

→ Este caso ocorre quando nem o caso (1) nem o caso (2) ocorrem.



De um modo geral, é provável que os IEDs existentes no mercado só disponibilizem arquivos oscilográficos para *download* quando os mesmos já tenham sido completamente gravados no repositório “público” do IED. Entretanto, como o funcionamento do repositório de arquivos não é mencionado na norma, o token FTJ pode ser usado para tratar esta situação patológica. Para isso, basta configurá-lo com o tempo máximo que uma oscilografia leva para ser gerada/gravada pelo IED. No exemplo de CNF dado, foi usado um valor de 3 segundos, considerando que o tempo máximo que o IED pode ficar incrementando um mesmo arquivo é 3 s.

17.5.4 Serviços MMS implementados

Para viabilizar a implementação deste novo processo, os seguintes serviços MMS foram implementados:

```
FileOpen  
FileClose  
FileDirectory  
FileRead
```

17.5.5 Monitoração de Mensagens Físicas

Para monitorar o tráfego das PDUs dos serviços acima, o usuário deve considerar um pseudo conversor de protocolo nomeado f61850:

```
mmf    f61850    rx
```

17.5.6 Remoção de Arquivos Repetidos

A estrutura de controle do processo ftp61850 inclui informações essenciais para que o processo seja capaz de decidir se um arquivo encontrado no IED deve ser lido ou não, assim como a maneira como ele deve ser gravado no repositório local. Considerando que essa estrutura é mantida em memória volátil, caso haja uma interrupção do funcionamento do processo e posterior reinício, o processo não terá como tomar nenhum tipo de decisão baseada em arquivos lidos anteriormente. Nesse caso, o processo iniciará seu funcionamento considerando todos os arquivos encontrados no IED como arquivos novos e fazendo o download de todos eles.

Se este procedimento gerar arquivos repetidos no repositório local (caso de IEDs que não apagam seus arquivos após terem sido transferidos) deve-se usar o *script* “remove_iguais” que remove arquivos iguais, recursivamente a partir do diretório corrente, preservando em um grupo de iguais o primeiro deles determinado por critério alfabético.