FileHash, ICH e AICH

Filehash

eMule (in particolare dalla versione 0.50a) identifica i file in modo univoco nel network tramite una tripletta di informazioni: la dimensione, il FILEHASH e il ROOT HASH, calcolati con algoritmi matematici partendo dalle dimensioni e dal contenuto del file, e **non dal nome**. Questo assicura che anche file con lo stesso nome vengano in realtà visti come differenti da eMule, essendo diverso il filehash.

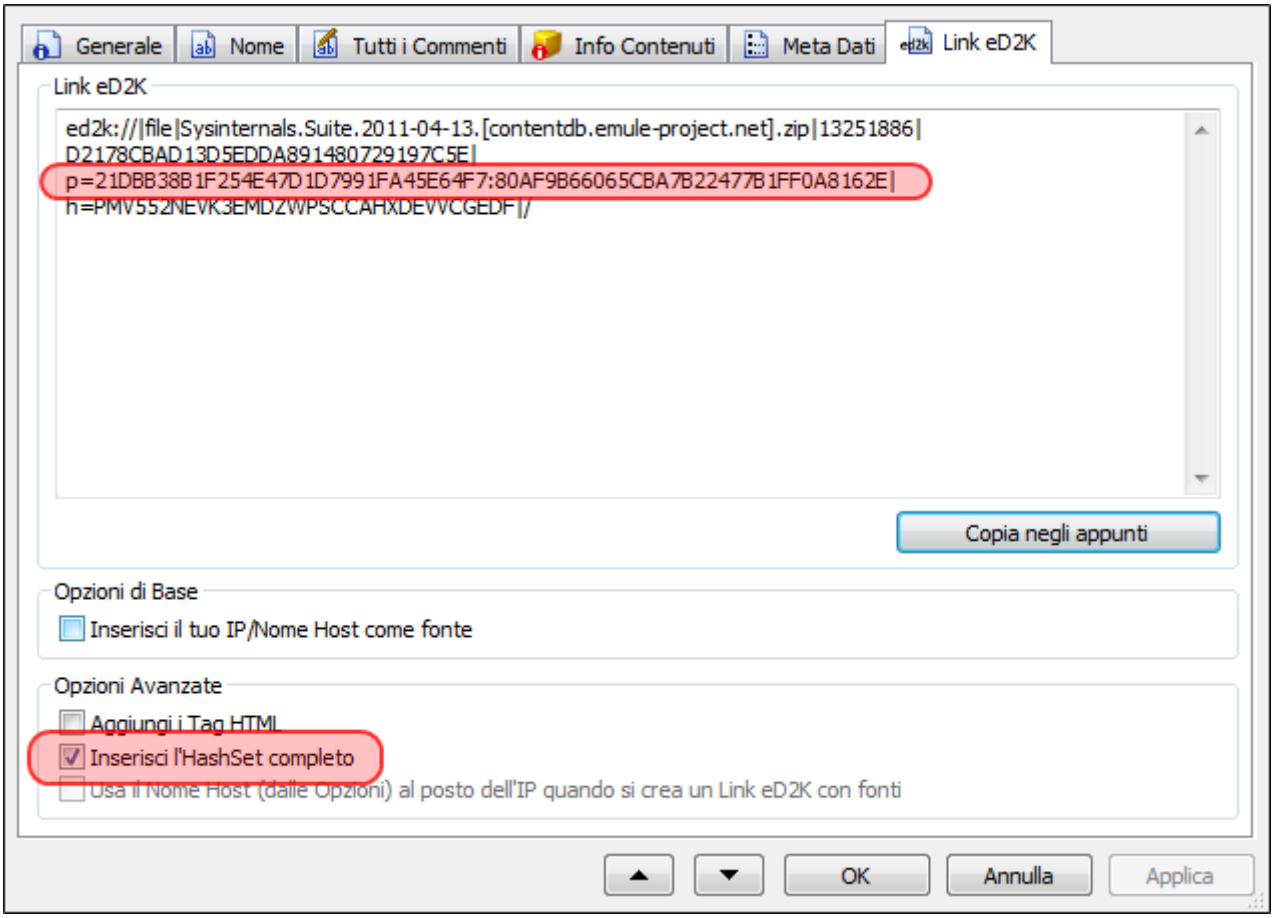
Potete conoscere il FILEHASH dei vostri file dalla finestra **File Condivisi** di eMule, nella colonna **ID File**.

Quando il vostro eMule comunica la lista dei file che condividete ai server, alla rete Kademlia oppure ad altri utenti, utilizza sempre il filehash per ogni singolo file. Come avrete notato leggendo il contenuto della colonna ID File, la complessità del filehash garantisce che, almeno statisticamente, la probabilità di trovare in rete due file diversi ma con stesso filehash è praticamente zero!

ICH e AICH

eMule scarica ed invia i file in piccoli blocchetti da **180KB** e fino ad un massimo di **9.28MB**, cioè la singola parte di un file.

Per ogni singola parte il programma calcola il relativo **PartHash** attraverso l'algoritmo MD4, così da ottenere *l'HashSet* finale composto dalla sequenza dei singoli PartHash. E' possibile inviare un link eD2K contenente l'**HashSet completo** attraverso l'omonima opzione, dove con **P=** è indicato l'hash MD4 delle singole parti separate tra loro dal carattere **:** (due punti).

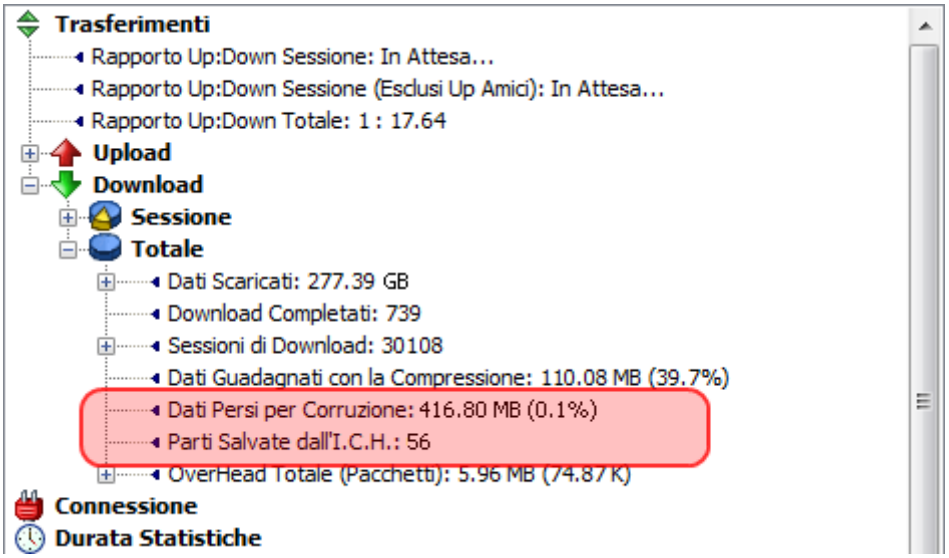


A partire dall'HashSet delle singole parti il programma è in grado di calcolare l'hash MD4 totale del file, ovvero il **FileHash**, dal quale può infine verificare la correttezza dei file trasferiti e gestire il recupero delle parti corrotte.
Può capitare infatti che durante il trasferimento o il salvataggio sul proprio Hard Disk una o più parti subiscano una qualche alterazione, rendendo di fatto il file scaricato diverso dal file desiderato, e quasi certamente corrotto.

Per impedire tale alterazione eMule adotta dei meccanismi di **verifica** della coerenza dei dati, ed eventualmente di **recupero** delle parti incoerenti. Quello principale è l'**ICH** (Intelligent Corruption Handling, tradotto “Trattamento intelligente delle corruzioni”).

Per ogni parte scaricata l'ICH verifica la corrispondenza con il relativo PartHash e se risulta differente allora elimina la parte e ne richiede nuovamente il trasferimento. Il sistema è definito “**Intelligent**” perché in prima battuta richiede solo il trasferimento dei primi 180KB del chunk e ricontrolla quindi l'hash di tutto il chunk da 9,28MB per vedere se l'errore è stato corretto. Se risulta ancora differente, allora itera il procedimento e richiede il trasferimento del successivo blocchetto da 180KB, verificando nuovamente la corrispondenza dell'Hash dell'intero chunk.
Nel migliore dei casi, cioè se il blocchetto corrotto era il primo, allora l'ICH recupera l'intero chunk dopo aver trasferito solo i primi 180KB. Nel peggiore dei casi, cioè se il blocchetto corrotto era quello finale, l'ICH deve invece riscaricare completamente il chunk. Mediamente l'ICH trasferisce nuovamente il 50% del chunk, quindi circa 4.5 MB.

Nella finestra delle Statistiche vengono riportati i dati relativi al numero di successi dell'ICH e la percentuale di dati persi per corruzione:



Dal Registro Eventi nella finestra Server invece è possibile controllare quando e come l'ICH interviene per recuperare le corruzioni. Esempio potreste leggere una serie di messaggi simili:

```
La parte scaricata n°26 è corrotta :(  (NomeFile)"
I.C.H.: Recuperata parte corrotta n°26 (NomeFile), Risparmiati: 10.54 KB
```

> Se notate una successione di messaggi relativi alla corruzione della stessa parte, vuol dire che probabilmente non ci sono parti buone in condivisione ed eMule continua a scaricare sempre la stessa parte corrotta. Si consiglia di pazientare inizialmente ma dopo qualche giorno, se il recupero di quella parte fallisce ancora, eliminate il file e cercatene un altro.

Quando la corruzione interessa gli ultimi byte del chunk da 9.28MB, l'ICH è costretto quindi a scaricare quasi interamente il chunk anche se solo una piccola parte è realmente necessaria, creando quindi uno spreco di banda.
Per ovviare a tale inefficienza dalla versione 0.44 è stato introdotto l'**AICH** (Advanced Intelligent Corruption Handling).

Questo sistema utilizza un approccio più “fine” e consiste in un set di hash “costruiti” a partire non dalle singole parti da 9.28MB, ma direttamente dagli Hash dei singoli blocchetti da **180KB** (chiamati **BlockHash**) e raggruppati in una struttura “ad albero” che va a formare il **Root Hash**.

Ora se eMule sta scaricando un file e rileva una parte corrotta chiederà ad una fonte scelta in modo random un **pacchetto di recupero** contenente l'intero HashSet AICH, cioè tutti i BlockHash dell'intero chunk (più ulteriori hash di verifica). A pacchetto ricevuto avvierà il confronto tra i BlockHash calcolati sulla parte corrotta e i BlockHash contenuti nel pacchetto di recupero ricevuto, quindi da un confronto “uno ad uno” sarà in grado di individuare immediatamente il singolo blocchetto da 180KB corrotto, e richiederne quindi un nuovo trasferimento.

Rispetto all'esempio precedente dell'ICH nel caso peggiore, con l'AICH non sarà necessario scaricare l'intero chunk per recuperare la corruzione sul blocchetto finale, ma sarà sufficiente scaricare direttamente gli ultimi 180K senza dover trasferire nuovamente l'intero chunk, portando quindi ad un notevole risparmio di banda.

Nel Registro Eventi verrà mostrato un avviso simile:

```
L'AICH ha recuperato con successo 1.56 MB di 9.28 MB dalla parte n°15 per
NomeFile
```

L'AICH è una funzione addizionale rispetto al classico ICH, quindi qualora eMule non riuscisse a recuperare un Hashset AICH valido, potrà sempre tentare il recupero delle parti corrotte con il vecchio ICH.

Per Inguaribili Curiosoni

FileHash

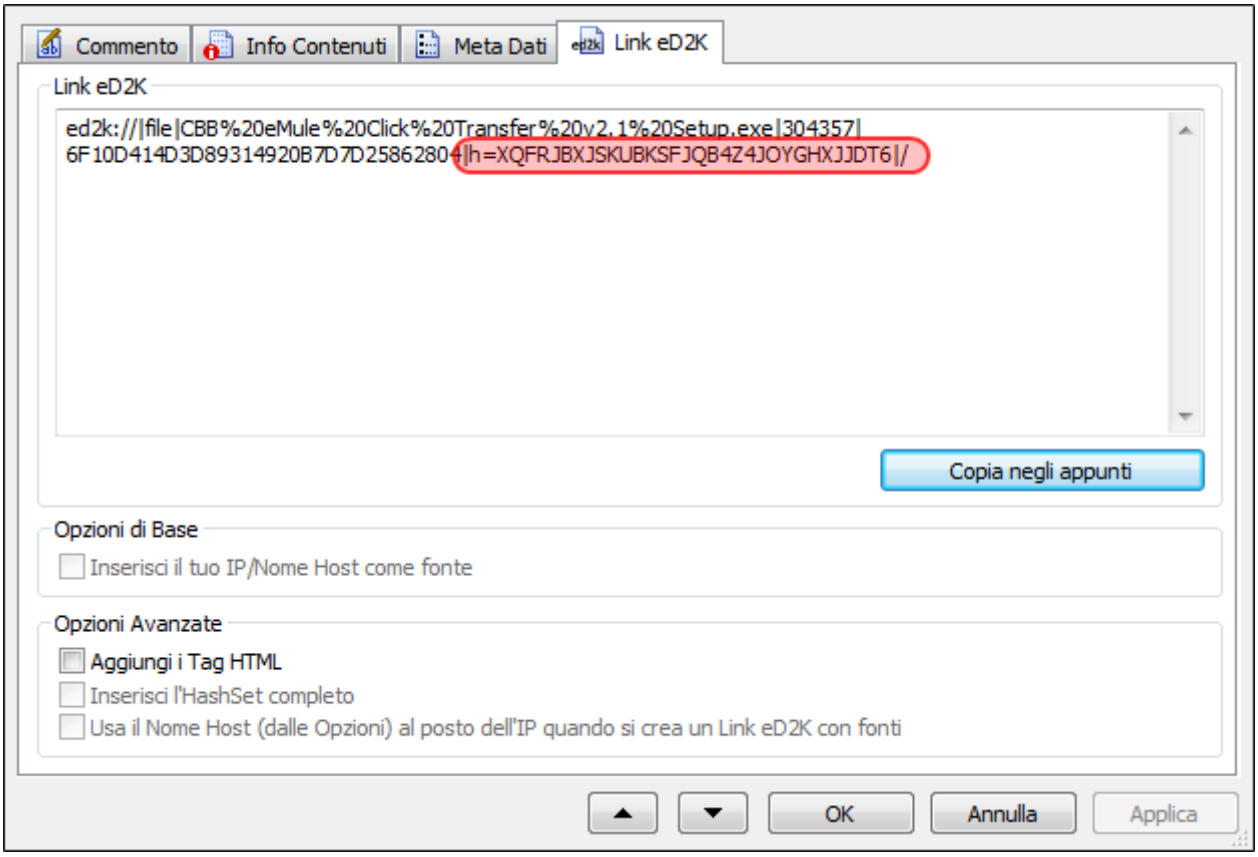
eMule ha sempre usato il FileHash per identificare univocamente un file in rete a partire dal suo contenuto. Il FileHash è calcolato a partire dalla sequenza di Hash MD4 dei chunks del file, cioè a partire dai singoli **PartHash**.
Questo significa che per un file di 15 MB, il FileHash è l'hash MD4 della stringa composta da due PartHash, cioè dall'hash MD4 del primo pezzo di 9.28 MB e del restante pezzo da 5.72 MB, in questo modo:

```
hash della prima parte + hash della seconda parte =
stringa

e8c636d0c0486378bf61e6a3000d0fb7 + 16f39bc5c0f6d73f2578cca229590fa1 =
e8c636d0c0486378bf61e6a3000d0fb716f39bc5c0f6d73f2578cca229590fa1

hash della stringa =
FileHash

MD4('e8c636d0c0486378bf61e6a3000d0fb716f39bc5c0f6d73f2578cca229590fa1') =
ea4bbab715bab849a7ecb22f506e5500
```

Va da sè che se il file è più piccolo di 9.28 MB, allora il FileHash sarà semplicemente l'hash MD4 del file completo.

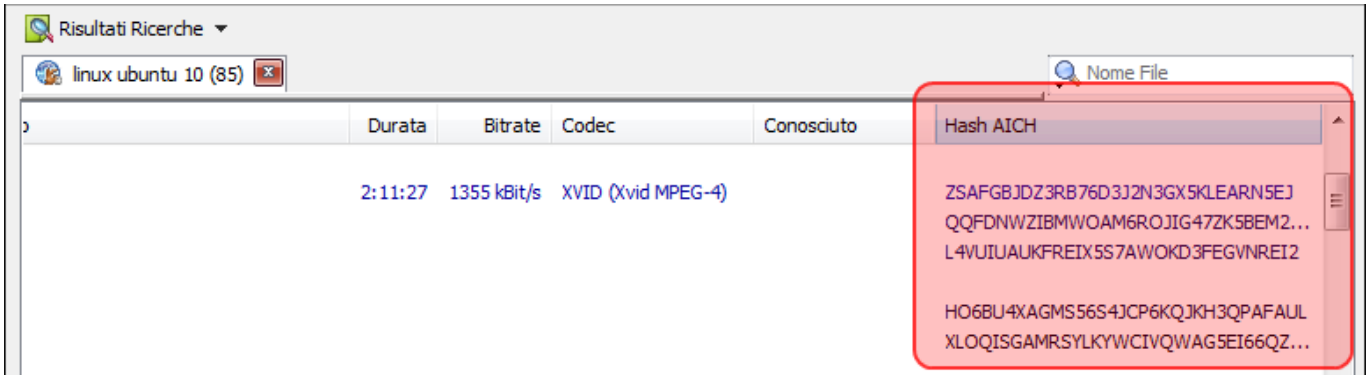
Root Hash

eMule, dalla versione 0.50a, utilizza il Root Hash di default nei link ed2k. E' quello segnalato dalla lettera h, seguita da un segno di uguale, seguito da una serie di 32 cifre esadecimali. Quei 32 esadecimali sono, per l'appunto, il Root Hash.

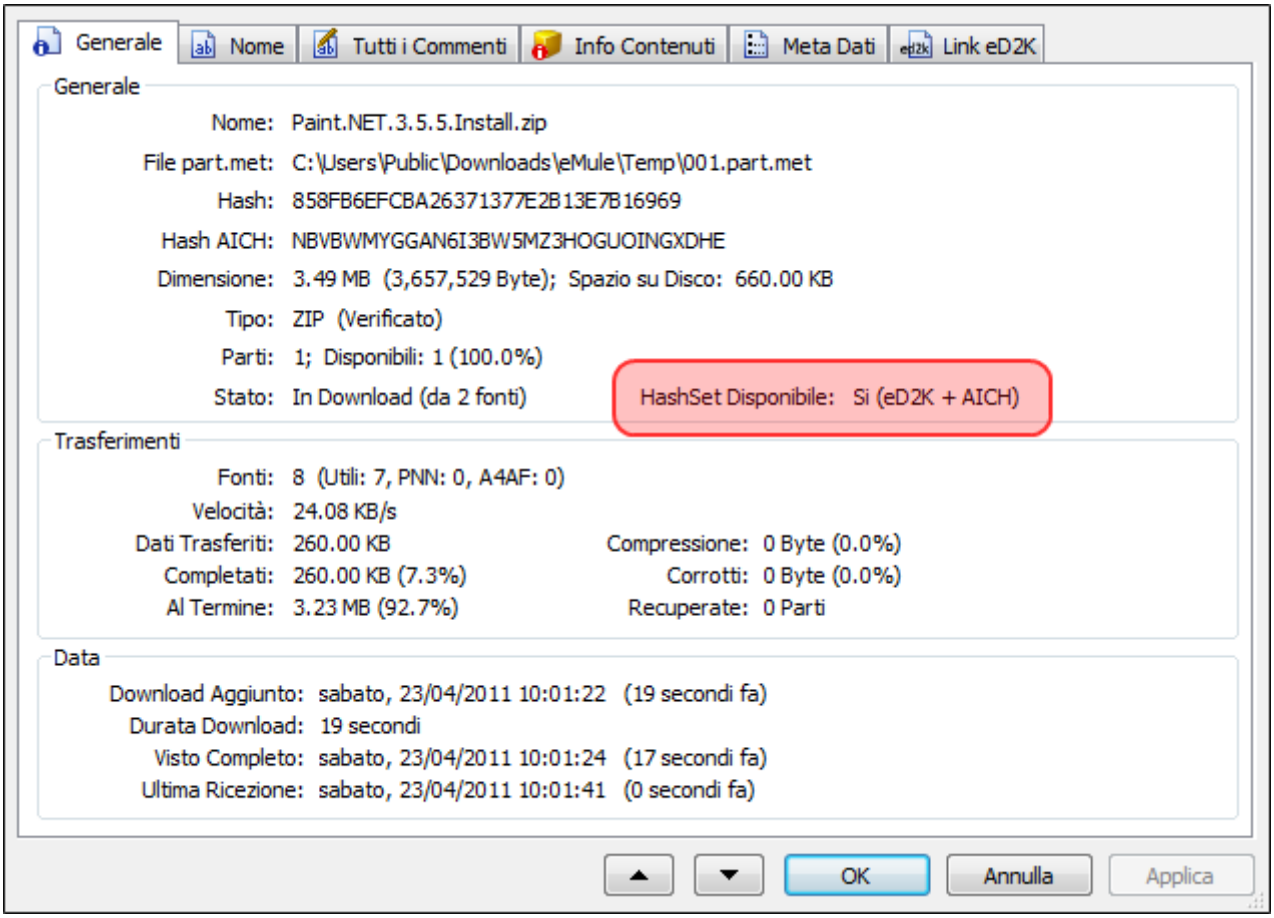


Il Root Hash è calcolato a partire dall'AICH Hashset (qui sopra ne abbiamo parlato in termine di “albero di hash”) dell'intero file. Per un singolo chunk (9,28 MB) l'AICH Hashset è costituito da 53 hash SHA-1 calcolati per ognuno dei 53 “blocchetti” da 180 KB, che sono l'unità minima recuperabile dall'AICH.

Sempre dalla 0.50a il Root Hash potrebbe essere disponibile se il file proviene da una ricerca su rete Kad (un motivo in più per utilizzarla!). Se la ricerca individua **solo un Hash AICH** per file, e se tale hash proviene da **almeno 1/3** delle fonti, allora l'hash viene ritenuto valido esattamente come il root hash allegato al link eD2K, e sarà visibile se abilitate la colonna **Hash AICH**. Non tutti i file verificheranno le due condizioni, quindi molti file potrebbero esserne sprovvisti:



Quindi con le ultime versioni chi riceverà il link o sfrutterà la ricerca via rete Kad, sarà sicuro di poter contare sull'AICH per il recupero delle parti corrotte. Potete verificare nelle informazioni sul file alla voce **Hashset Disponibile** se il file è provvisto anche dell'hashset AICH:



Se il link eD2K o la ricerca su rete Kad non forniscono un Root Hash, eMule può comunque recuperarlo dalle fonti complete del file ma lo considera affidabile solo a due condizioni:

- Almeno **10 indirizzi IP** unici devono averci inviato lo stesso valore di hash;
- Almeno il **92% delle fonti** totali del file hanno un hash che corrisponde con quello inviatoci.

Questo hash verrà comunque posto ad un livello di fiducia più basso e sarà valido solo nella sessione corrente, cioè non verrà né salvato e né diffuso, e non sarà quindi possibile ricreare un link AICH partendo da questo hash.

Non appena eMule avrà ricostruito l'intero HashSet AICH, cioè il file sarà completo, provvederà a memorizzarlo nel file **known2_64.met** e inizierà a diffonderlo ai client che ne faranno richiesta.

Si capisce bene che, soprattutto i *releaser*, dovrebbero sempre fornire il link eD2K completo di Root Hash perché essendo il file raro risulterebbe impossibile recuperarlo dalle altre fonti. Il problema comunque è secondario in quanto, come detto precedentemente, con le ultime versioni il link eD2K ha il Root Hash già incluso.

Legenda

- Chunk = Part = 9.28MB (9728000 bytes) = “pezzo”
- Block = 180KB = “blocco”
- FileHash = “Hash del file”
- RootHash = “Hash AICH”
- PartHash = “Hash del Part”

- BlockHash = “Hash del blocco”

From:
http://www.emule-wiki.org/ - **eMule-Wiki**

Permanent link:
http://www.emule-wiki.org/it/guide/advanced/filehash_ich_aich

Last update: **2013/08/28 23:50**