

V.L.A.D.

Vampire Locates and Acquires Data

Versione 1.00

di Rodolfo Calzetti (postmaster@rudyz.net)

A partire da un albero di espressioni regolari, **VLAD** analizza un file di testo e ne estrae dati popolando un contenitore strutturato denominato **\$BLOOD**. Espressioni regolari e criteri di estrazione dei dati sono in un file descrittore dipendente dal formato del file. Come esempio ecco un descrittore per acquisire il formato *CBI* (remote banking):

```
{
  "root":{
    "match":"^ 61",
    "matchtype":"B",
    "reset":"0",
    "levels":{
      "testata":{
        "match":"^ 61(.{7}).{42}(.{5}).{5}(.{12})(. {3})(. {6})(. )(.{15}).*$",
        "matchtype":"A",
        "reset":"1",
        "method":"SET",
        "attributes":{
          "BANCA":{"index":"2"},
          "CONTO":{"index":"3"},
          "DIVISA":{"index":"4"}
        }
      },
      "dettaglio":{
        "match":
          "^ 62(.{3}).{7}(.{6})(. {6})(. {1})(. {15})(. {2}).{2}(.{16}).{25}(.{34}).*$",
        "matchtype":"B",
        "reset":"0",
        "method":"SET",
        "indexes":"",
        "attributes":{
          "DATAVAL":{"index":"2", "type":"D", "format":"DMY", "output":"Ymd"},
          "DATABAN":{"index":"3", "type":"D", "format":"DMY", "output":"Ymd"},
          "SEGNO":{"index":"4"},
          "IMPORTO":{"index":"5", "type":"N", "sepdec":",", "minus":"D"},
          "CAUSALE":{"index":"6"},
          "DESCRIZIONE":{"index":"8"},
          "REGISTRO":{"index":"0", "remove":"1"}
        }
      },
      "levels":{
        "registro":{
          "match":"^ 63.{10}(.*)$",
          "matchtype":"A",
          "reset":"0",
          "method":"SET",
          "indexes":"",
          "attributes":{
            "REGISTRO":{"index":"1", "conflict":"OLD.' '.NEW"}
          }
        }
      }
    }
  }
}
```

VLAD applica la *REGEX* del livello radice per suddividere in blocchi (blocchi di primo livello) il file. Ciascun blocco viene suddiviso in sottoblocchi (blocchi di secondo livello) dal primo sottolivello del livello radice. Se il sottolivello ha a sua volta sottolivelli, il procedimento viene iterato su ciascun blocco di secondo livello e così via.

Terminata l'analisi dei blocchi di primo livello con i parametri del primo sottolivello del livello radice, si passa a rianalizzare i blocchi di primo livello col secondo sottolivello del livello radice. Il procedimento termina quando l'ultimo blocco di primo livello è stato analizzato dall'ultimo sottolivello del livello radice.

Ogni volta che viene analizzato un blocco, per prima cosa si estraggono i dati secondo quanto è specificato con il parametro **"attributes"**.

In simboli, siano

- **L[0]** il livello radice
- **B[0]** l'intero file
- **L[0, 0], L[0, 1], ...** i sottolivelli di **L[0]**
- **B[0, 0, 0], B[0, 0, 1], ...** i sottoblocchi di **B[0]** ottenuti con **L[0, 0]**
- **B[0, 1, 0], B[0, 1, 1], ...** i sottoblocchi di **B[0]** ottenuti con **L[0, 1]**
- ...
- **L[0, 0, 0], L[0, 0, 1], ...** i sottolivelli di **L[0, 0]**
- **B[0, 0, 0, 0], B[0, 0, 0, 1], ...** i sottoblocchi di **B[0, 0, 0]** ottenuti con **L[0, 0, 0]**
- ...

Da ciascun **B[... , k]**, *VALD* estrae i dati secondo il parametro *attributes* del livello **L[...]**. Il modo in cui *\$BLOOD* viene popolato dipende dagli altri parametri di livello.

Elenchiamo di seguito tutti i parametri che si possono presentare in un file descrittore.

"root":	livello radice.
"levels":	vettore dei sottolivelli.
"match":	espressione regolare relativa al livello.
"matchtype":	individuazione del blocco: ciò che viene trovato dalla <i>REGEX</i> è "A" tutto il blocco, "B" l'inizio fino alla prossima occorrenza o <i>EOF</i> .
"method":	"SET" (i dati vengono travasati in corrispondenza degli indici), "PUSH" (i dati vengono appesi in corrispondenza degli indici)
"reset":	cancella i dati di <i>\$BLOOD</i> in corrispondenza degli indici specificati.
"clear":	cancella tutti i dati in <i>\$BLOOD</i> .
"indexes":	indici di livello separati da virgole; il simbolo "#" viene sostituito con l'ultimo indice caricato in corrispondenza degli indici che precedono.
"csv":	il blocco viene convertito da <i>CSV</i> prima di essere analizzato.
"enabled":	abilita l'esecuzione del livello: di default è 1 (<i>TRUE</i>).
"inverted":	la logica di esecuzione viene invertita: se la <i>REGEX</i> non trova nulla i sottolivelli vengono eseguiti sullo stesso blocco.
"attributes":	sezione che definisce sorgente e destinazione dei dati per ogni blocco.

"remove":	se 1 (<i>TRUE</i>) il dato viene tolto da <i>\$BLOOD</i> .
"index":	è l'indice dell'operatore parentesi della <i>REGEX</i> in corrispondenza del dato da estrarre.
"target":	nome del dato di destinazione in <i>\$BLOOD</i> .
"default":	valore di default del dato non trovato.
"type":	tipo di dato: "S" (stringa), "N" (numero), "D" (data).
"sepdec":	simbolo separatore dei decimali per dati numerici.
"minus":	simbolo del segno meno per dati numerici.
"format":	formato in ingresso per le date; possibili valori sono: "YMD", "DMY", "MDY", "Y#M#D", "D#M#Y", "M#D#Y"
"output":	formato stringa in uscita per le date (convenzioni <i>PHP</i>).
"conflict":	comportamento da adottare se una variabile da registrare è già presente in <i>\$BLOOD</i> ; per esempio, il valore "OLD." '.NEW' indica che la vecchia stringa va concatenata colla nuova mediante uno spazio; il valore "OLD." + '.NEW' indica che vecchio valore numero va sommato al nuovo; "OLD" preserva il vecchio; con "NEW" il vecchio viene sovrascritto.

Un oggetto *\$VLAD* scatena **eventi** prima e dopo l'analisi di ciascun blocco; per esempio, se un livello ha come nome "dettaglio", verranno chiamate le funzioni

```
dettaglio_prepare($match)
```

```
dettaglio_complete($match)
```

qualora esistano: *\$match* è il pacchetto dati che verrà aggiunto a *\$BLOOD*.

I **metodi** utilizzabili di *\$VLAD* sono i seguenti.

Carica un descrittore da file:

```
$VLAD->config(' ../../customize/ryvlad/cbi.json');
```

Arricchisce le opzioni del descrittore con altri valori in formato *JSON*:

```
$VLAD->config( '{"source":" ../../customize/_import/fonte.cbi"}' );
```

Inizia l'algoritmo di analisi.

```
$VLAD->bleed();
```

I **metodi** utilizzabili di *\$BLOOD* sono i seguenti.

Accesso ai dati in lettura:

```
$BLOOD->data(indici, nome);
```

Inserisce/modifica un pacchetto di dati in corrispondenza degli indici specificati:

```
$BLOOD->set(indici, array( "NOME" => "Pippo" ) );
```

Appende un pacchetto di dati in corrispondenza degli indici specificati:

```
$BLOOD->push(indici, array( "NOME" => "Pippo" ) );
```

Restituisce il numero di pacchetti dati in corrispondenza degli indici specificati:

```
$BLOOD->count(indici);
```

Cancella tutti i dati:

```
$BLOOD->clear();
```

Cancella i dati in corrispondenza degli indici specificati:

```
$BLOOD->reset(indici);
```

Rimuove un dato:

```
$BLOOD->remove(indici, nome);
```



Quest'opera di RODOLFO CALZETTI (<http://www.rudyz.net>) e le idee ivi contenute sono state rilasciate con licenza
CREATIVE COMMONS ATTRIBUZIONE - CONDIVIDI ALLO STESSO MODO 3.0 ITALIA.

Per leggere una copia della licenza visita il sito web <http://creativecommons.org/licenses/by-sa/3.0/it/> o spedisce una
lettera a Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.