SpamassNotFirstHop

I'm using spamass-milter and the -notfirsthop rules like DYNABLOCK don't work properly

Spamass-milter 0.2.0 includes a new feature that constructs a pseudo Received-header to provide SpamAssassin with additional useful data. Since incoming email is passed to the milter prior to SendMail adding its Received-headers, spamass-milter constructs a Received-header itself to pass to SpamAssassin so that SpamAssassin can take the information into consideration. This header is only used by SpamAssassin as spamass-milter does not send this header back to Sendmail. Sendmail continues to add its own header as it normally does. The following line is the code in spamass-milter that send the pseudo header to Spamassassin:

```
assassin->output((string)"Received: from "macro_s"("smfi_getsymval(ctx," ")") by "smfi_getsymval(ctx,"j")"; "macro_b"\r\n");
```

macro_b is only available to spamass-milter if defined in Sendmail's configuration; in many situations it's not defined. As a result macro_b gets set to an empty string, but there is a space prior to macro_b which remains. The end result is a whitespace preceding the newline characters. In SpamAssassin's Received.pm we have the following:

```
my @rcvd = ($hdrs =~ /^(\S.+\S)$/gm);
```

This code populates `@rcvd` with the headers that will be passed to `parse_received_line()`. As you can see from the regex, it doesn't take kindly to whitespaces before the newline characters. As a result the pseudo-header from spamass-milter never gets parsed and SpamAssassin never knows that the sending MTA existed. This causes problems since the assumption is made that the Received-header added by the receiving MTA has already been added before SpamAssassin processes the email.

This results in behavior such as header checks (including RBL checks) not being performed on the sending MTA. This causes -notfirsthop rules like DYNABLOCK to be practically useless since it will not catch open proxies on cable/dsl machine which deliver directly to the SpamAssassin mail server. In addition rules like DYNABLOCK will generate false-positives on emails from dialup and cable/dsl users which travele through a single hop (such as their ISP's mail server) before reaching SpamAssassin since SpamAssassin will see only a single header (the one added by the ISP) and will believe that the mail has been delivered directly to the mail server from the user's machine.

This bug is currently (as of July 2004) fixed in the CVS version of spamass-milter, but no official release including the fix has been released at this time. To fix this you can either use the CVS version of the spamass-milter code or modify the code in spamass-milter.cpp to remove macro_b (since it's not terribly important) as shown below:

```
assassin->output((string)"Received: from "macro_s" ("smfi_getsymval(ctx," ")") by "smfi_getsymval(ctx,"j")";
\r\n");
```