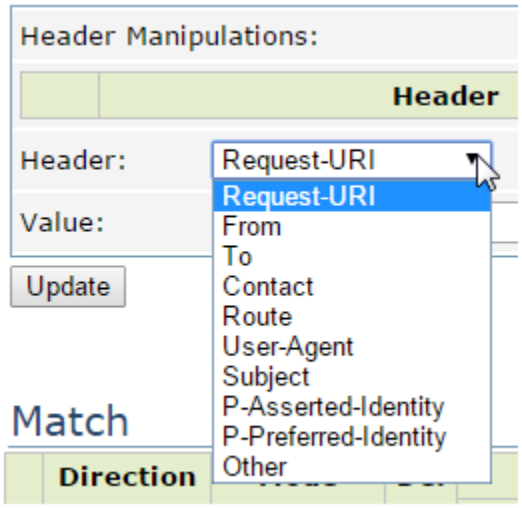
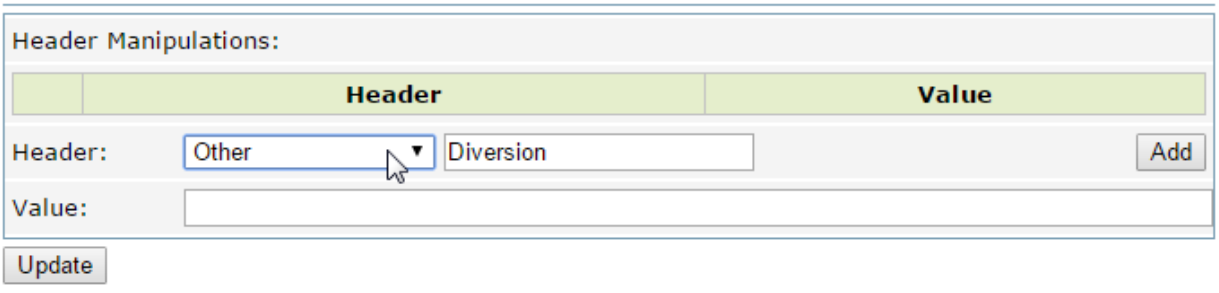
HMRs

HMR stands for "**Header Manipulation Rules**" and is something you would want to create in an EdgeMarc for achieving interoperability between a pbx and a provider for certain calls to work. You see some pbxs are limited and different in how they send their headers, it could be these headers are not acceptable to the provider when being sent out so this could result in the provider rejecting the call. The solution to this is actually having the EdgeMarc change these headers on the pbxs behalf and "regenerating" the header so that when the provider receives it, it accepts the call. The EdgeMarc does this on the B2BUA page and allows the change of specific headers like From, To, etc. Not all pbxs need this, but some may depending on how different their outgoing headers are. The following shows how these hmrs are added. First of all when you click the option list of headers in an EM you will then see a specific list of headers.



If you see the header that you want to change here, then perfect. If not you must select "**Other**" and enter the name of the header in the input box exactly as you want it to appear in the SIP message.



In above example, "**Diversion**" was entered.

When entering a value, the first thing to know is when you are entering a value for a header, the header value is entered as a series of segments where each segment is separated by a **+** sign.

```
segment + segment + segment + ...
```

Segments are one of 3:

1.) Literal: These are literals that are "literally" copied as is character for character and inserted into the header. They are separated by single quotes like this **'sip:'** and anything you write will appear in the header. So if you write **'Excellent'** then this will appear in the header. Just copies from value field and pastes into header.

2.) Variable: These are system predefined values. They are called variables because they are "extracted" from the "received" SIP message, then pasted into that specific field. They are always preceded with **$** like**$from.uri.user** and separated by **+**.. This example "extracts" the user name portion of the URI from the received message, then pastes it into that field. This is important list of variables:

HMR Variables:

Header Name	Variable	Remarks
From	$from.dispname	Display name
	$from.uri.user	user name portion of the URI
	$from.uri.host	host portion of the URI
	$from.uri.port	port from the URI
	$from.uri.parameter[user]	Value of URI Parameter 'user'
	$from.uri.header[h1]	Value of URI header 'h1'
	$from.parameter	Value of From parameter 'tag'
To	Same as From	
Contact	Same as From	
Route	Same as From	
Record-Route	Same as From	
P-Asserted-Identity (PAI)	Same as From (use $pai instead of $from)	
P-Preferred-Identity (PPF)	Same as From (use $ppf instead of $from)	
Via	$via.ip.host	Value of sent-by
	$via.ip.port	Value of sent-by port
	$via.parameter[branch]	Value of parameter 'branch'
Request URI	$request.method	SIP method
	$request.uri.user	Request-URI user
	$request.uri.host	Request-URI host
	$request.uri.port	Request-URI port
Other Headers	$diversion.text	Extract the entire diversion header from the received message

Environment Variables

Variable Name	Value
$env.in_intf_host	IP Address of the interface on which the message was received
$env.in_intf_port	TCP/IP Port on which the message was received
$env.out_intf_host	IP Address of the interface from which the message will leave the system
$env.out_intf_port	Source port that will be used for sending the message
$env.target_host	IP Address of the destination of the message
$env.target_port	Destination port of the message
$env.target_domain	Domain name of the destination of the message
$env.available_host	IP Address of the currently available/reachable configured SIP server
$env.available_port	Destination port of the currently available/reachable configured SIP server
$env.available_domain	Domain name of the currently available/reachable configured SIP server

3.) Function: Expressions that perform a specific operation with variables. For example you can "replace" a value in a received message to something you want instead, like replacing a +1 with nothing when leaving the EM. This is done like this **replace($to.uri.user, '+1', '')** , this example replaces the **+1** in the to uri portion with **nothing**, effectively deleting it. This example is useful if you want to modify E.164. The main function is replace and here is its format.

replace(src_string, 'search_string', 'replace_value')

replace() function searches the "src_string" for all occurrences of "search_string" and replaces them with "replace_value".

Example:

Search all occurrences of "88" in the diversion header and replace them with "408".

```
replace($diversion,'88','408')
```

Another function is a substring function. This function essentially doesn't 'replace' a specific digit(s) but rather 'removes' a digit or digits from a 'specific' position within the number.

substr (src_string, start_index, len)

substr() function extracts a sub-string of length "*len*" from the src_string starting at index "*start_index*". The index is zero based, so, for example, the first character is index "0", and the second character is index "1". If the start_index is negative, counting starts from the end of the string. For example "-1" would mean the last character, and "-2" would mean the 2nd to last character. If "*len*" is zero this means the entire string. This can be useful when stripping characters from the left side of "*src_string*".

Examples:

Strip leading "+1" from the username in From header

```
'<sip:' + substr($from.uri.user, 2, 0) + '@' + $env.target_domain + '>'
```

To summarize, the '**start_index**' are the first number of digits starting from the left and the '**len**' which is 0 here means remove entire string. Its very important that you know when to use the **substr** function and the **replace**function. The replace function is meant to replace not to delete. You can use the replace function to delete like with this example that deletes '+1' and replaces it with 'nothing':

replace($to.uri.user, '+1', '') , however this only works in this example because +1 only exists at the beginning of the number. So if you dialed **+1408-555-1155** then it would change it to **408-555-1155** and would work. However lets say you changed it to **replace($to.uri.user, '1', '') ,** this says replace every 1 digit. which create problems because if you dialed **1408-555-1155** then it would change it to **408-555-55** and delete 'all' 1 digits. You can see the difference!

EXAMPLE DEMONSTRATIONS:

NOTE: The following require **<>** around their value: (black means standard & always should be there).

FORMAT:

From: Requires **<>**
from name <sip:DID@IP-Address:Port>

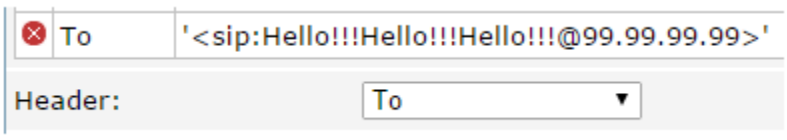
To: Requires **<>**
to name <sip:DID@IP-Address:Port>

Contact: Requires **<>**
contact name <sip:DID@IP-Address:Port>

Request-URI: **<>** NOT Required
sip:DID@IP-Address:Port

1.) Literals-

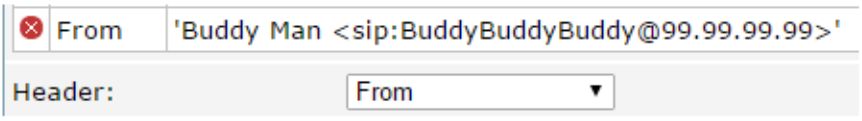
Lets assume for now that you want to change a header that is already in the list above. Let us change the To header with a **Literal**. I am going to select '**To**' then enter the following value:



Now lets see what happens when we submit this and a call is made.

```
<<<<<<<<<<<<<<<<sip header stop<<<<<<<<<<<<<<<<<<<
 [tos 0xb8]
17:45:18.185256 209.92.109.7.5060 > 70.167.106.233.5060:
>>>>>>>>>>>>>>>>>sip header start>>>>>>>>>>>>>>>>>>>
INVITE sip:6782344113@70.167.106.233:5060 SIP/2.0
Via: SIP/2.0/UDP 209.92.109.7:5060;branch=z9hG4bK58027346
From: FN258 LN258 <sip:6782384035@docsis.cox.net>;tag=3az
da2d3e
To: <sip:Hello!!!Hello!!!Hello!!!@99.99.99.99>
```

Very interesting indeed. It copied it exactly as we entered it between single quotations, hence a literal. Lets try it with From:




```
18:03:04.613389 209.92.109.7.5060 > 70.167.106.233.5060:
>>>>>>>>>>>>>>>>>sip header start>>>>>>>>>>>>>>>>>>>
INVITE sip:6782344113@70.167.106.233:5060 SIP/2.0
Via: SIP/2.0/UDP 209.92.109.7:5060;branch=z9hG4bK2023791648
From: Buddy Man <sip:BuddyBuddyBuddy@99.99.99.99>;tag=3az08g6Rsxdj70tla7A2247B47
aa46bb
To: <sip:Hello!!!Hello!!!Hello!!!@99.99.99.99>
```

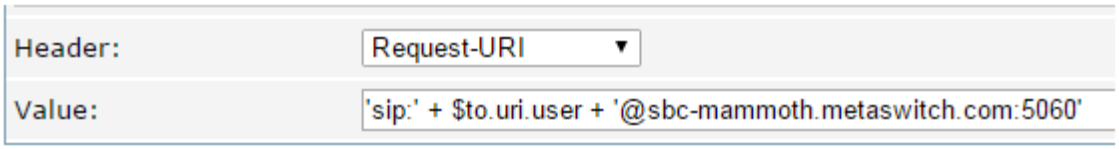
That's basically how you enter a literal & as you see it just copies & pastes.

2.) Variables-

Lets now do an example for variables. Lets assume we want to change the Request-URI. This is how it will look before we make a change:

```
INVITE sip:6782344113@70.167.106.233:5060 SIP/2.0
```

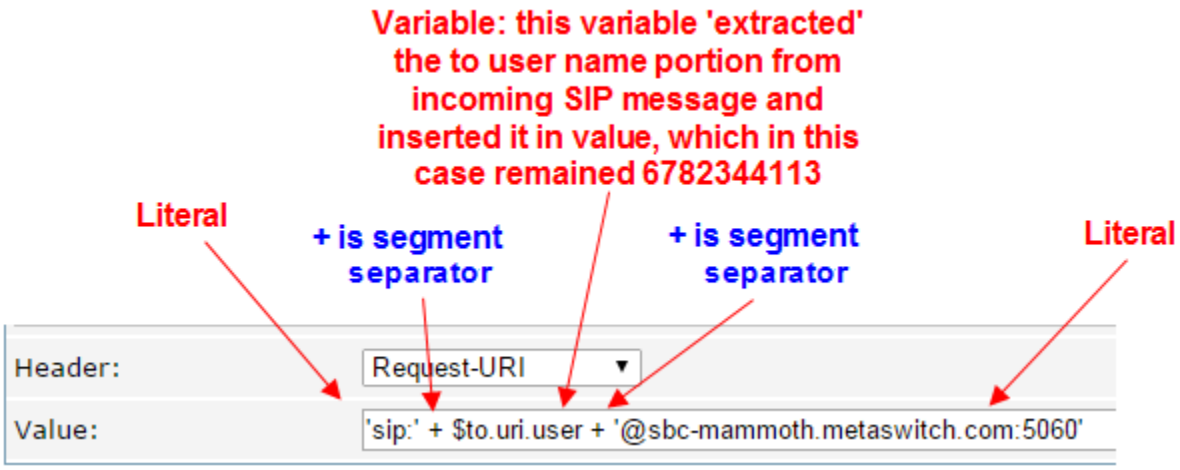
Now lets say I want to change the domain above into a different domain yet keep other fields as is. This is what I would do:



And after submitting this is what it shows:

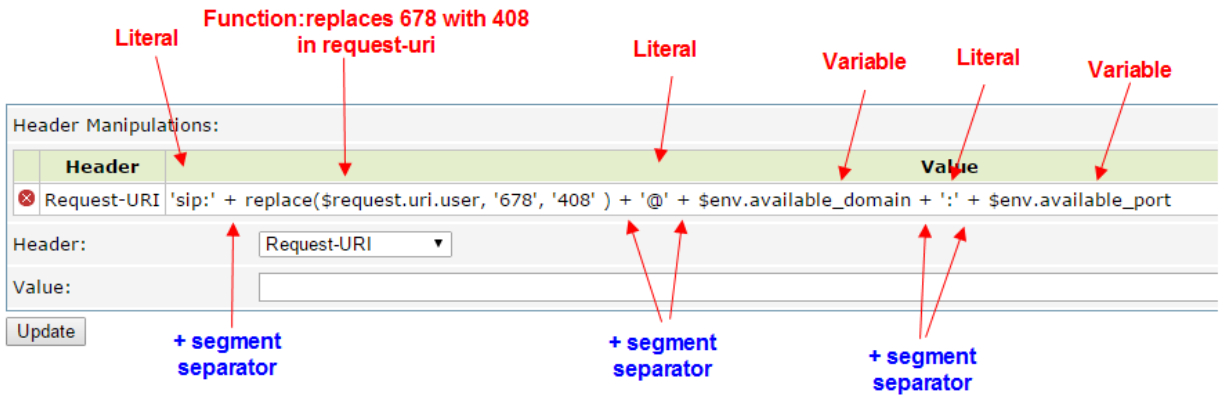
```
<<<<<<<<<<<<<<<sip header stop<<<<<<<<<<<<<<<<<<<
 [tos 0xb8]
17:24:11.149505 209.92.109.7.5060 > 70.167.106.233.5060:
>>>>>>>>>>>>>>>>>sip header start>>>>>>>>>>>>>>>>>>>
INVITE sip:6782344113@sbc-mammoth.metaswitch.com:5060 SIP/2.0
```

As you can see it only replaced the domain as suspected. Lets analyse what happened:



3.) Functions-

Lets check out an example for functions. Lets do a replace function together. Let us replace the outgoing DID number area code and change it from 678 to 408. This is what we do:



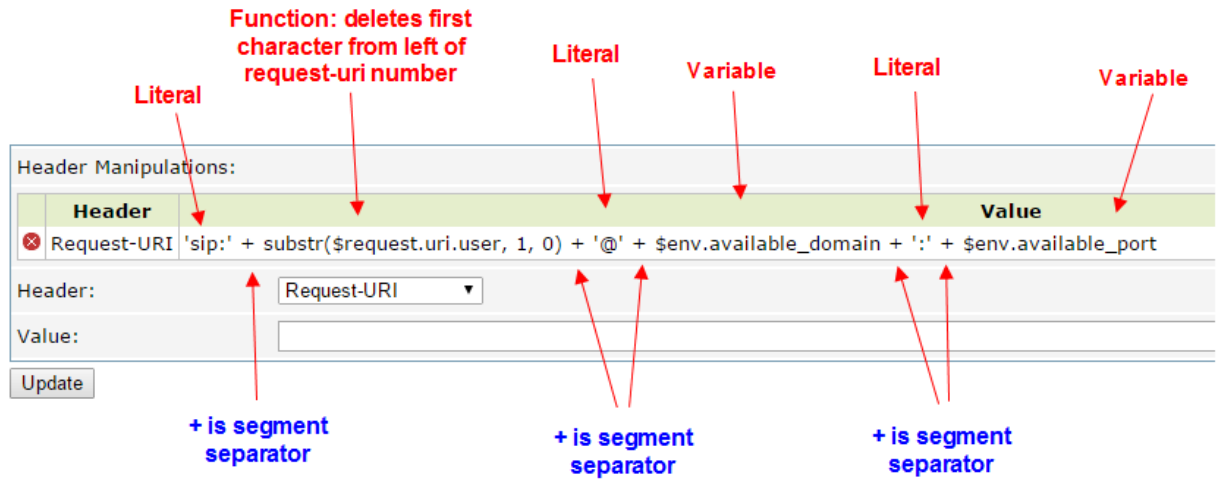
Lets see now what the EM does when it receives a dialed number from the pbx starting with 678.

```
18:35:54.963704 10.10.33.11.5060 > 10.10.33.1.5060:
>>>>>>>>>>>>>>>>sip header start>>>>>>>>>>>>>>>>>>>
INVITE sip:6782344113@10.10.33.1 SIP/2.0
Via: SIP/2.0/UDP 10.10.33.11:5060;branch=z9hG4bK-4591f83a
From: FN258 LN258 <sip:6782384035@10.10.33.1>;tag=6f1b644270ddee98o5;re
To: <sip:6782344113@10.10.33.1>
Call-ID: f8d98d0a-f0049c70@10.10.33.11
CSeq: 101 INVITE
Max-Forwards: 70
Contact: "Edgewater" <sip:6782384035@10.10.33.11:5060;ref=6782384035>
Expires: 240
User-Agent: Cisco/UC320W-2.3.2(6)
Allow-Events: talk, hold, conference, x-spa-cti
Content-Length: 395
Allow: ACK, BYE, CANCEL, INFO, INVITE, NOTIFY, OPTIONS, REFER
Supported: x-sipura, replaces
Content-Type: application/sdp

<<<<<<<<<<<<<<<<sip header stop<<<<<<<<<<<<<<<<<<<
 [tos 0xb8]
18:35:54.994780 209.92.109.7.5060 > 70.167.106.233.5060:
>>>>>>>>>>>>>>>>sip header start>>>>>>>>>>>>>>>>>>>
INVITE sip:4082344113@docsis.cox.net:5060 SIP/2.0
Via: SIP/2.0/UDP 209.92.109.7:5060;branch=z9hG4bK428539021
```

As shown it replaced the area code to 408 in INVITE going to the provider. Cool!

Lets try an example with **substr** function. Lets delete the first character to the left of a number in request-uri, lets strip the 1 in 16782344113. This is what we add:



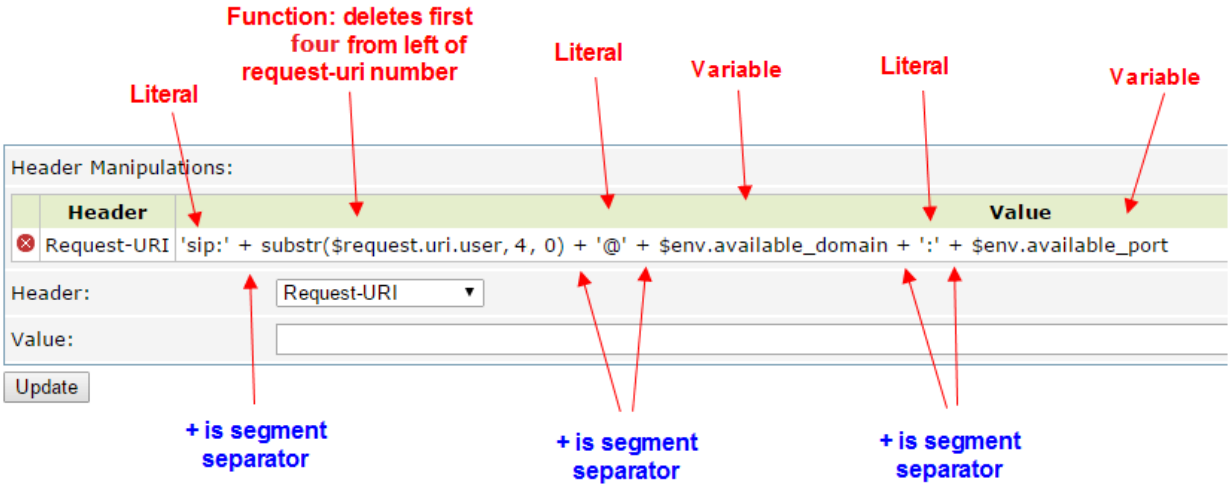
This is what happens when call is made:

```
>>>>>>>>>>>>>>>sip header start>>>>>>>>>>>>>>>>>>
INVITE sip:1678234411 3@10.10.33.1 SIP/2.0
Via: SIP/2.0/UDP 10.10.33.11:5060;branch=z9hG4bK-e6d5245b
From: FN258 LN258 <sip:6782384035@10.10.33.1>;tag=ccdf3577cf8cb0e4o5;ref=258
To: <sip:16782344113@10.10.33.1>
Call-ID: 29bc8333-9f0b5e58@10.10.33.11
CSeq: 101 INVITE
Max-Forwards: 70
Contact: "Edgewater" <sip:6782384035@10.10.33.11:5060;ref=6782384035>
Expires: 240
User-Agent: Cisco/UC320W-2.3.2(6)
Allow-Events: talk, hold, conference, x-spa-cti
Content-Length: 395
Allow: ACK, BYE, CANCEL, INFO, INVITE, NOTIFY, OPTIONS, REFER
Supported: x-sipura, replaces
Content-Type: application/sdp

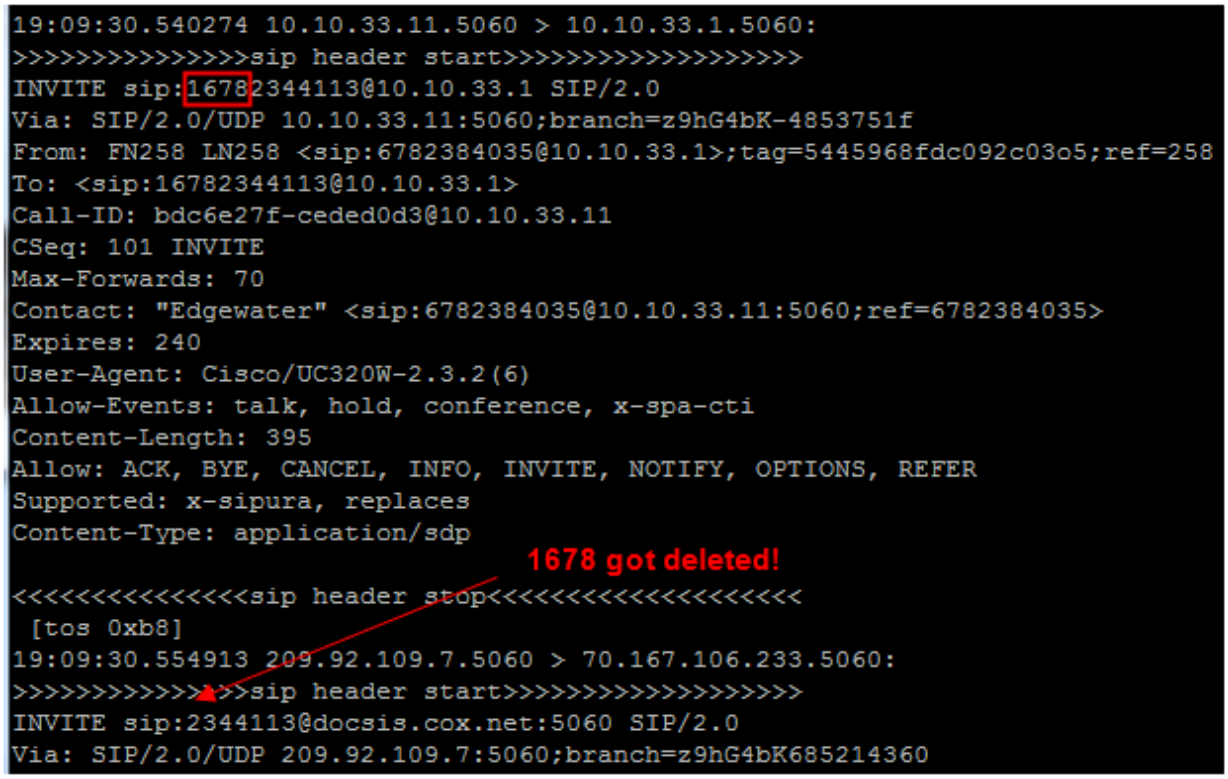
<<<<<<<<<<<<<<<sip header stop<<<<<<<<<<<<<<<<<<
 [tos 0xb8]
18:56:49.370774 209.92.109.7.5060 > 70.167.106.233.5060:
>>>>>>>>>>>>>>>sip header start>>>>>>>>>>>>>>>>>>
INVITE sip:6782344113@docsis.cox.net:5060 SIP/2.0
Via: SIP/2.0/UDP 209.92.109.7:5060;branch=z9hG4bK34773555
```

Got deleted!

That's great, it deleted the 1 of the INVITE going to the provider. Lets say though we wanted to delete the first 4 digits from the left, this is what we would enter:



Lets see what happens when we make the same call:



As shown the first four digits from left were deleted.

This concludes the HMR tutorial. I hope you had fun.