

Preventing Access to Anonymous Proxy Servers

Introduction

The explosive growth of publicly available anonymous proxy websites have become a major issue for organisations who employ web access management software (like *getbusi*) to control, filter and manage their Internet access. Anonymous proxy websites are a problem because they allow an organisation's users to effectively bypass their web access management system's filters and controls, providing unrestricted, untraceable and unmonitored access to Internet content. Since building an anonymous proxy website is a trivial task for anyone with rudimentary computing skills and broadband Internet access, traditional list-based filtering methods cannot stay ahead of the increasing number of proxy sites.

How do users use anonymous proxy websites to bypass getbusi?

There are many types of proxies used for various reasons. In fact, *getbusi* implements a proxy server to manage Internet access. The majority of anonymous proxies that users access to bypass their *getbusi* server's filtering are categorised as *circumventors*. A circumventor is a web-based page with a form field allowing visitors to enter and retrieve a URL that would otherwise be blocked by their *getbusi* system. Circumventor sites take advantage of the likelihood that their domain or IP address will not present in list filters, and therefore not be blocked. They retrieve the prohibited website and present it in a sub-frame on their own web page. Since the prohibited website's URL will be displayed in the GET variables required by the form field (which would trigger the *getbusi*'s filters), they encrypt those GET variables to bypass filtering software. These types of circumventors are also known as CGI proxies, because they typically use CGI or PHP to implement the proxying functionality.

A second type of circumventor rapidly gaining popularity is one that implements SSL (Secure Sockets Layer) tunnels to deliver prohibited content. These circumventors take advantage of the same security technology that online banking and other eCommerce websites use. Assuming the circumventors aren't themselves being blocked, users are able to establish a secure connection and request the prohibited content. The content is delivered through the secure tunnel to the requesting user's browser. Since the *getbusi* system has no way to inspect the data within the secure connection, it cannot filter or otherwise manage the data being transmitted.

Two methods to combat circumventing proxies

As previously indicated, filter-based blocking of circumventor proxies is clearly ineffective. Circumventor sites are easy and cheap to set up. Many circumventor sites don't even have a registered domain name; they are accessible by IP address, making them even more difficult to locate and filter. Since filtering cannot hope to solve the problem, another method to restrict access to circumventor sites is required. Since there are two predominant types of circumventor sites, two methods to combat them are needed. To address these types of circumventors, *getbusi* in collaboration with one of our clients, have identified two new preventative measures. A third method has also been developed to specifically address unregistered home-based circumventors that are only accessible by IP address.

The first method helps combat CGI/PHP circumventing proxies by taking advantage of a common attribute: encrypting the GET variable of the requested website. When users first access a CGI/PHP circumventor, they are presented with a web-based form. Users enter the URL in the form field of the circumventor and click a button. The circumventor retrieves the web page and displays it in an inline frame, but it also displays the encoded URL of the requested resource. Since the encoded URL is readable by the *getbusi* server, an expression list can be created to filter out the encoded URL. Of the many CGI/PHP circumventor proxies tested, several common encoded expressions were found. In a test at one of our client sites, these expressions have proven to be quite effective in blocking CGI/PHP circumventor proxies.

The second method helps combat SSL-based circumventor sites by port blocking. Port blocking is also achieved in an expression list, with the difference that the list will block a port, rather than examining the data stream for a known expression. Although it is technically possible for the *getbusi* server to break SSL encryption and examine the data stream, it is inadvisable because all SSL encrypted websites (including online banking and other eCommerce sites) would be compromised. Therefore, the solution is to introduce an additional expression list to block a port (specifically, port 443), and then only allow access to valid secure sites with a good list. Although this method is inconvenient for some of your users while you build a list of valid secure sites to allow, it is the only method currently known to block SSL-based circumventor sites. Please consult your Users Guide to learn more about selectively allowing sites that are blocked by filters.

The third method involves blocking any request to an IP address. This will block those users who set up a circumventing proxy on their home machine without registering a domain, in the hopes that their personal circumventor will evade filtering software. This is also implemented as an expression list.

The combination of the three aforementioned expression list methods is the only way to proactively block circumventing proxies. *Getbusi* does not guarantee that these methods will block 100% of the available circumventing proxies, but early results are promising. Please also remember that there are legions of people inventing new ways to bypass filtering software.

Implementation

To implement the three methods to block circumventing proxies, you must first create expression lists. Then you must apply the newly created expression lists to one or multiple policies. Please note that you must first update your *getbusi* software through your web access management console. Only the latest version of *getbusi* supports the functionality contained in this document. For information on how to update your *getbusi* software, please consult your Users Guide.

Creating Query Blocking Expression Lists for CGI/PHP Circumventors

In order to implement an expression filter to block PHP/CGI based circumventing proxies, first open a browser to your *getbusi* Web Access Management Console. In the left hand column, click on the *Filter lists* link. Then click on *Expression lists*.

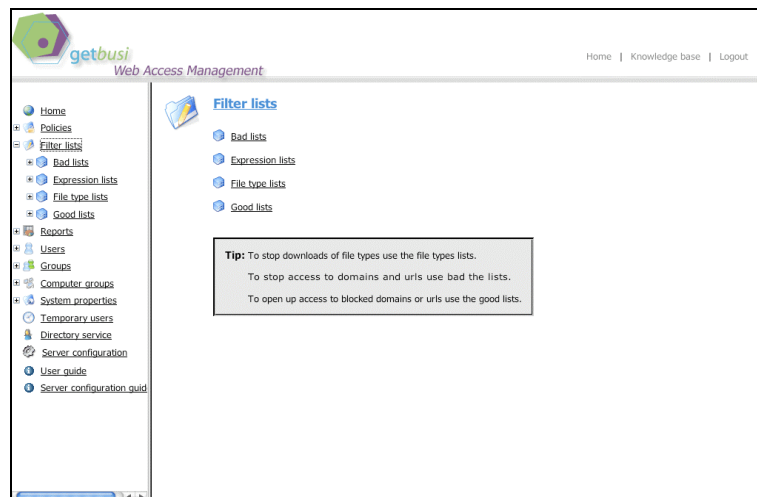


Figure 1

When the page refreshes, add the name of your proxy expression filter in the text field and click the green **Add** button. The example pictured in *Figure 2* shows the creation of an expression list named: proxies.

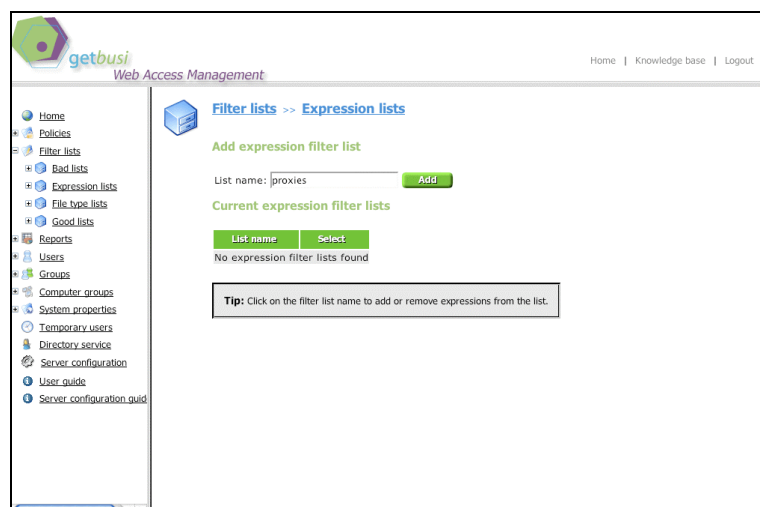


Figure 2

The new list will appear in the *Current expression filter lists* table. Click on the newly added list name to add expressions to the list.

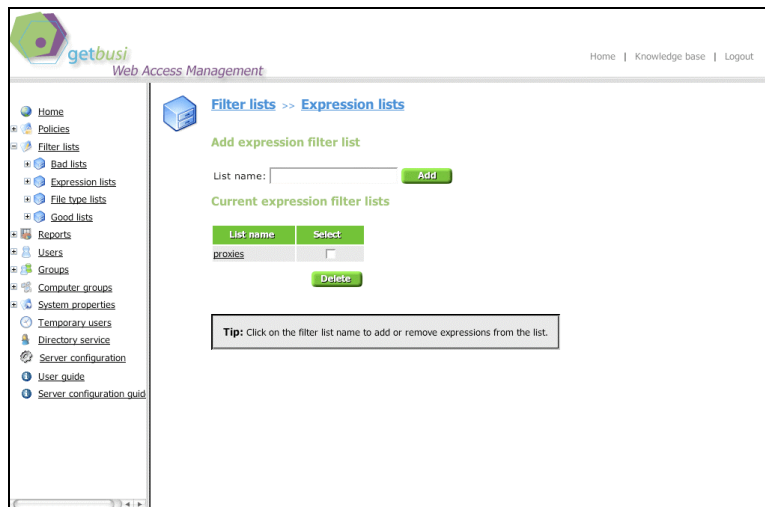


Figure 3

There are four expressions you'll need to add to your list: aHR0c, 68747, 7777, d3d3L. Enter each individually into the *List name* text box. Ensure that *Query block* is selected from the adjacent drop down box and click the green **Add** button. Note: the 0 in the expression aHR0c is a zero and not a capital O.

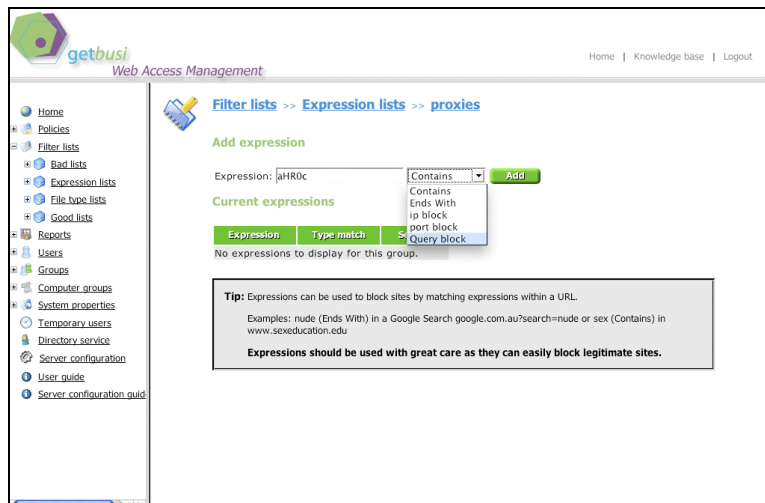


Figure 4

Figure 5 shows the completed expression list.

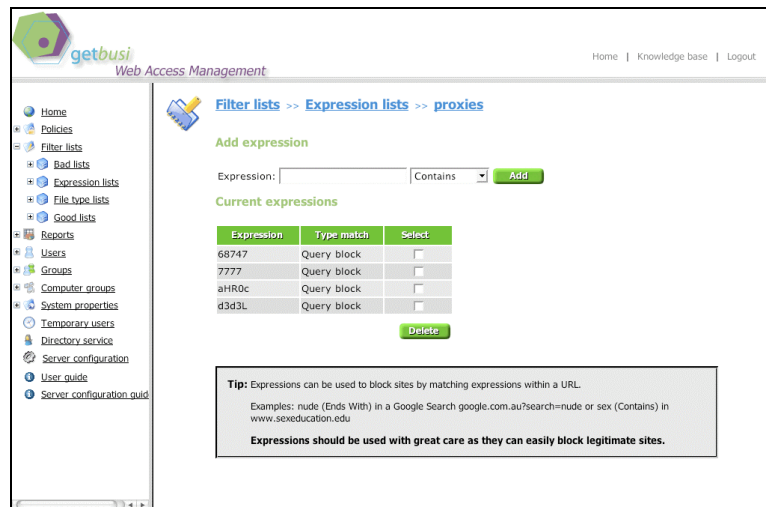


Figure 5

Creating Port Blocking Expression Lists for SSL Circumventing Proxies

The process of creating an expression list to block ports is very similar to the previous example. If you wish, you may add a port blocking expression to your CGI/PHP expression list. However, we recommend you create a separate list so that you can enable the port blocking expression separately from your CGI/PHP expression.

To add a port blocking expression list, navigate to *Filter lists* and then to *Expression lists*. In the *Add expression filter list* textbox, enter a new name for your expression list and click on the green **Apply** button. In the example, the port blocking list is named: *Port*. The new list will appear in the *Current expression filter lists* table.

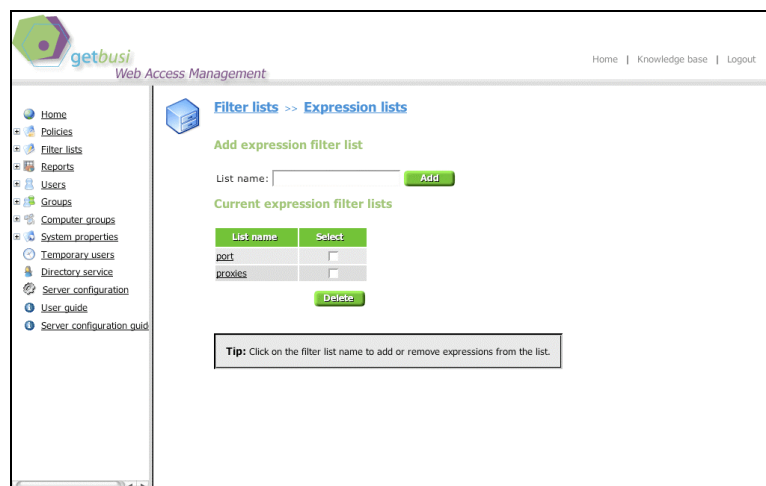


Figure 6

Click on the newly created list name to configure the filter. When the page refreshes, type *443* in the *Expression* text box. In the adjacent drop down box, select *port block*. Click on the green **Add** button to add the expression.

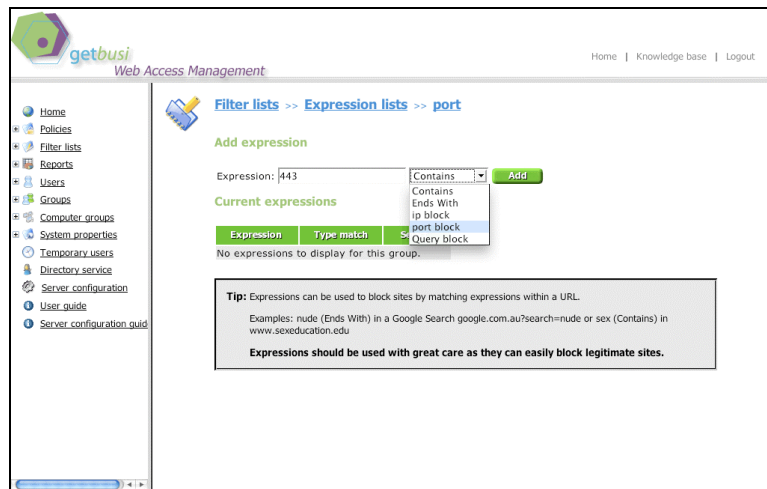


Figure 7

When the page refreshes, you should see the expression in the *Current expressions* table as shown in Figure 8.

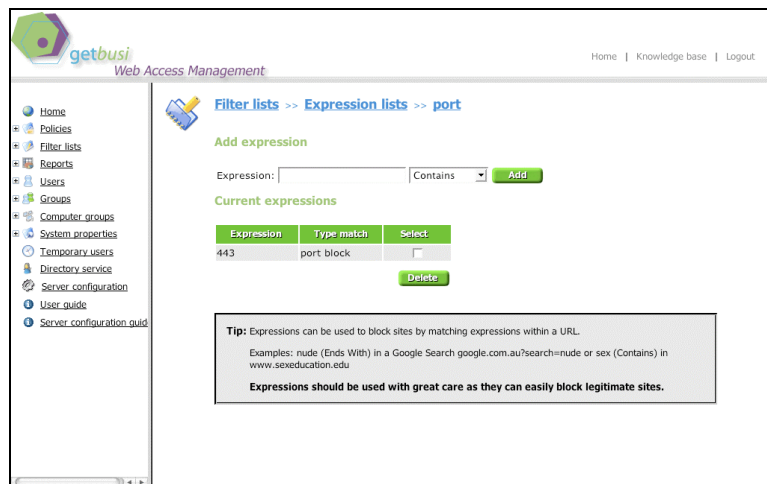


Figure 8

Blocking IP Address Requests

Blocking IP address requests does not specifically target circumventing proxies. It is a measure that can help in the effort, in that it will block any user requests to an IP address rather than a standard web address. However, since many circumventing proxies are home-built and served off of home ADSL connections, many of them will not have a registered domain and may only be reached by IP address. On the other hand, there are some (but not many) legitimate websites which may be served from IP addresses, or redirect to an IP address, which this filter will block. Because of this, you should setup a third expression list so that you may selectively activate or deactivate this expression from a policy.

To add an IP address blocking expression list, navigate to *Filter lists* and then to *Expression lists*. In the *Add expression filter list* textbox, enter a new name for your expression list and click on the green **Apply** button. In the example, the IP address blocking list is named: *IPAddress*. The new list will appear in the *Current expression filter lists* table.

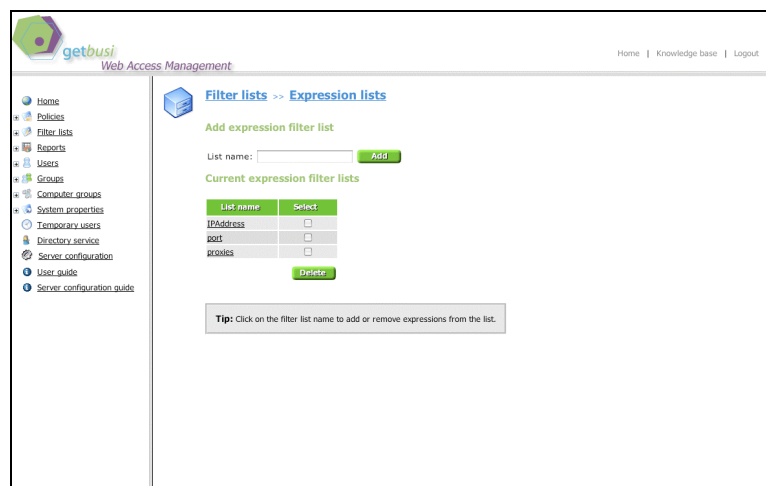


Figure 9

Click on the newly created list name to configure the filter. When the page refreshes, type the word *any* in the *Expression* text box. In the adjacent drop down box, select *IP block*. Click on the green **Add** button to add the expression.

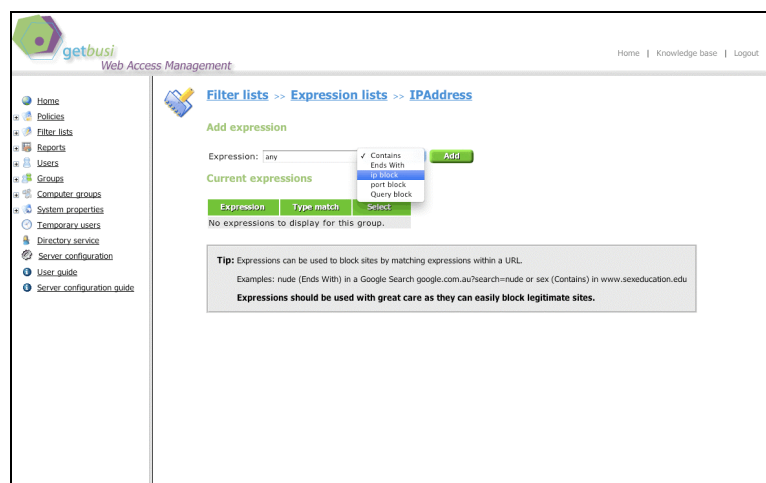


Figure 10

When the page refreshes, you should see the expression in the *Current expressions* table as shown in *Figure 11*.



Figure 11

This concludes the three types of expression filters to be added to help prevent access to circumventing proxies. The next step is to apply the newly created expression filters to one or more policies.

Applying Expression Lists to Policies

Once all of the expressions have been successfully added, the expression filter must be applied to a policy. To apply the expression filter to a policy, click on the *Policies* link in the left-hand column. When the *Policies* page refreshes, click on the policy in the *Current policies* table you wish to apply the expression filter to.

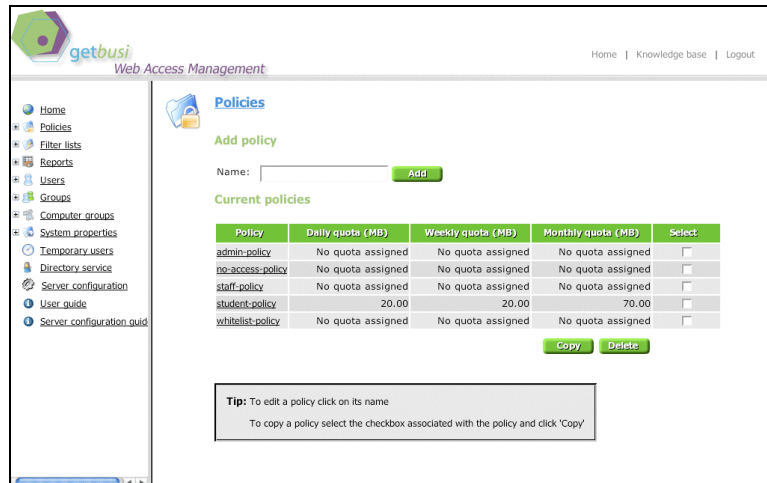


Figure 12

Once the page refreshes, the active policy will be displayed in the right-hand frame. Along the top, there are two rows of green tabs. By default, the *Quota* tab will be displayed for the policy. Click on the green *Expressions* tab to show available expression lists. Your newly created expression lists should appear in the table.

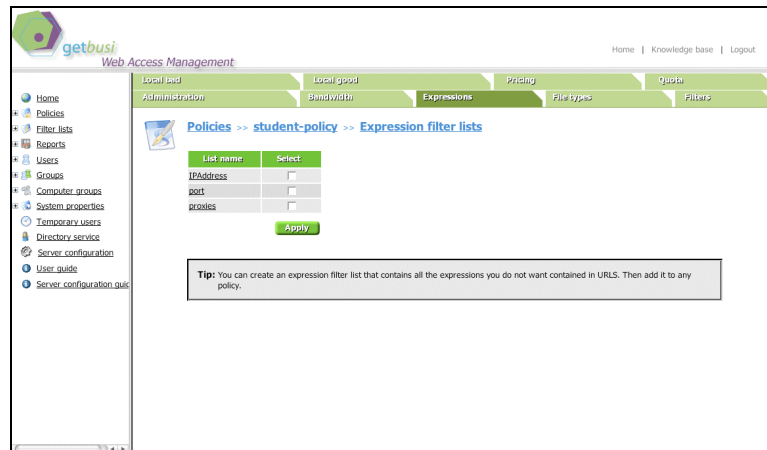


Figure 13

In the table listing available expression lists, locate your newly created expression lists. Check their corresponding checkboxes and click on the green **Apply** button. The expression lists are now applied to your policy.

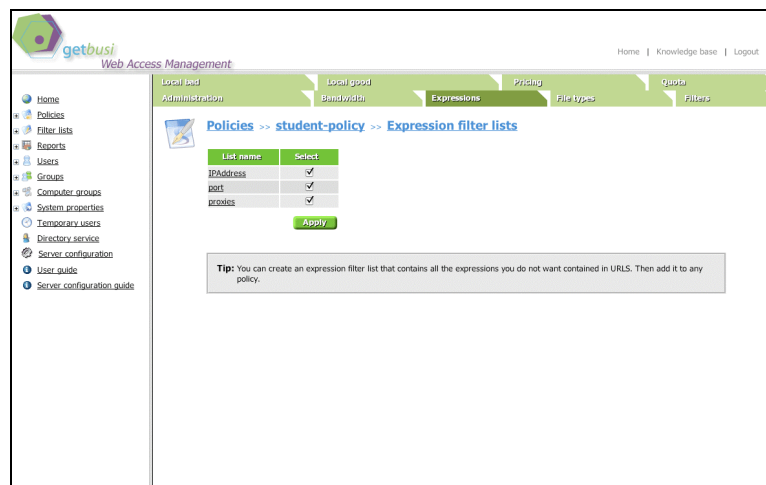


Figure 14

You may apply the same newly created expression lists to multiple policies, if you wish. Simply navigate to the policy, click on the *Expressions* tab in the policy, check each the expression list's checkbox and click on the green **Apply** button. You do not have to create separate (identical) expression lists for each policy. Existing lists may be applied to multiple policies.

Conclusion

As web access management systems get implemented in an increasing number of organisations, new methods to bypass or defeat them will be invented. At *getbusi* we are committed to providing our customers with the best and latest technology to manage their Internet access. We also rely on our customers to provide feedback and observations so that we can improve our product. If you have suggestions or comments regarding *getbusi*'s web access management product, please feel free to email us at: support@getbusi.com. Thank you for choosing *getbusi* as your web access management solution.