NativeSPInProcess

The <InProcess> element contains settings governing the portion of the SP that runs inside the web server. It also includes content specific to particular web servers that supply an inadequate native configuration mechanism. On version 2.4 and above, it is required for IIS usage, optional otherwise.

```
<InProcess logger="native.logger" checkSpoofing="true">
    <Extensions>
        <Library path="adfs-lite.so" fatal="true"/>
    </Extensions>

    <ISAPI/>

</InProcess>
```

When omitted, the default attribute values below are used, no extensions are loaded, and no IIS configuration information is supplied.

Attributes

- `logger` (local pathname) (default is `native.logger` on 2.4+)
    - This attribute points to a log4shib/log4cpp configuration file that defines in-process logging behavior, generally writing to the `native.log` file. If set, this overrides the `logger` property in the `<SPConfig>` parent element, but if omitted (defaulted), the parent property will take precedence.
- `catchAll` (boolean) (default is false)
    - If true, "global" exception handlers are used to trap crashes and other "uncontrolled" exceptions. This attempts to keep the web server process running if at all possible, but makes diagnosing bugs difficult, and can lead to unstable runtime behavior. Generally should be omitted (or set to false).
- `unsetHeaderValue` (string) (default is empty string)
    - When HTTP headers are used to supply exported attributes to applications, this value will be placed into headers corresponding to attributes that are not supplied, or that have no values. Defaults to an empty string (to represent a "null" value), but may be set to a an actual value, such as "UNKNOWN".
- `checkSpoofing` (boolean) (default is true)
    - If true, the SP will examine incoming headers supplied by the client and will fail the request if any of them match header names "controlled" by the SP. Do **NOT** disable this setting unless you have a strong reason to do so, as it may result in security vulnerabilities.
- `spoofKey` (string)
    - Optional "secret" value used to prevent false alarms from the `checkSpoofing` option. Web servers do not generally provide a reliable means of detecting whether a request is directly from a client or has been internally redirected/rewritten in some fashion.

> ⚠ **Preventing Header Spoofing**
>
> When using Apache, strongly consider porting applications to rely on environment variables in place of headers.
>
> If this is not possible, the `checkSpoofing` feature should be enabled. Server-side redirects, aliasing, and modules like mod_rewrite tend to cause false alarms because the SP detects headers that it itself has already created. To prevent this, the `spoofKey` setting enables a special header as a way of detecting whether the headers being examined came from the SP or the client. The idea is to make it difficult for a client to guess this value and spoof the key, which then bypasses the spoof detection code.
>
> On the Windows/IIS platform, a random key is automatically generated to ensure the detection feature works safely. On other platforms, you need to establish the `spoofKey` setting yourself. It should contain a suitably long, random value, and you MUST prevent the client from accessing any server-side scripts that might expose the key value through a dump of arbitrary request headers.

Child Elements

- `<Extensions>`
    - Specifies in-process extension libraries.
- `<ISAPI>`
    - Supplements the native IIS configuration with information critical to the operation of the Shibboleth ISAPI filter. Unneeded for non-IIS deployments.